COMPUTATIONAL METHODS FOR GEOCHEMICAL MODELLING

Applications to carbon dioxide sequestration

ALLAN MOREIRA MULIN LEAL

A dissertation submitted to the Department of Earth Science and Engineering, Imperial College London, in partial fulfilment of the requirements for the degree of Doctor of Philosophy

Imperial College London
Department of Earth Science and Engineering
Qatar Carbonates and Carbon Storage Research Centre

April 2014

I herewith certify that all material in this dissertation which is not my own work has been properly acknowledged.

Allan Moreira Mulin Leal

Copyright Declaration

The copyright of this thesis rests with the author and is made available under a Creative Commons Attribution Non-Commercial No Derivatives licence. Researchers are free to copy, distribute or transmit the thesis on the condition that they attribute it, that they do not use it for commercial purposes and that they do not alter, transform or build upon it. For any reuse or redistribution, researchers must make clear to others the licence terms of this work.

PUBLICATIONS

Peer-Reviewed Publications

Leal, A.M.M., Blunt, M.J., and LaForce, T.C. (2013). A robust and efficient numerical method for multiphase equilibrium calculations: Application to CO2–brine–rock systems at high temperatures, pressures and salinities. *Advances in Water Resources*, 62(C):409–430.

Leal, A.M.M., Blunt, M.J., and LaForce, T.C. (2014). Efficient chemical equilibrium calculations for geochemical speciation and reactive transport modelling. *Geochimica et Cosmochimica Acta*, 131:301–322.

Leal, A.M.M., Blunt, M.J., LaForce, T.C. (2014). A chemical kinetics algorithm for geochemical modelling. *Applied Geochemistry*. Invited for publication.

Conference Proceedings

Leal, A.M.M., Blunt, M.J., LaForce, T.C. (2012). A Numerical Method for Chemical Equilibrium Calculations in Multiphase Systems. In *ECMOR XIII–13th European Conference on the Mathematics of Oil Recovery*. Biarritz: EAGE.

Abstract

Geochemical modelling is fundamental for solving many environmental problems, and specially useful for modelling carbon storage into deep saline aquifers. This is because the injected greenhouse gas perturbs the reservoir, causing the subsurface fluid to become acidic, and consequently increasing its reactivity with the formation rock. Assessment of the long term fate of carbon dioxide, therefore, requires accurate calculations of the geochemical processes that occur underground. For this, it is important to take into account the major water-gas-rock effects that play important roles during the gas storage and migration. These reactive processes can in general be formulated in terms of chemical equilibrium or chemical kinetics models.

This work proposes novel numerical methods for the solution of multiphase chemical equilibrium and kinetics problems. Instead of adapting or improving traditional algorithms in the geochemical modelling literature, this work adopts an approach of abstracting the underlying mathematics from the chemical problems, and investigating suitable, modern and efficient methods for them in the mathematical literature. This is the case, for example, of the adaptation of an interior-point minimisation algorithm for the calculation of chemical equilibrium, in which the Gibbs energy of the system is minimised. The methods were developed for integration into reactive transport simulators, requiring them to be accurate, robust and efficient. These features are demonstrated in the manuscript. All the methods developed were applied to problems relevant to carbon sequestration in saline aquifers. Their accuracy was assessed by comparing, for example, calculations of pH and CO_2 solubility in brines against recent experimental data. Kinetic modelling of carbon dioxide injection into carbonate and sandstone saline aquifers was performed to demonstrate the importance of accounting for the water-gas-rock effects when simulating carbon dioxide sequestration. The results demonstrated that carbonate rocks, for example, increase the potential of the subsurface fluid to dissolve even more mobile CO_2.

Acknowledgements

I would like to express my deepest gratitude to Professor Martin Blunt for all his guidance and support throughout my studies at Imperial College. It was an honour to be supervised by such a brilliant and distinguished researcher, whose character, intelligence, and austerity, whenever necessary, allowed me to learn more than just science!

Many thanks to Dr. Tara LaForce for her guidance during the first year of my doctorate program. Everything I achieved would not have been possible if she did not believe in my potential when selecting me for a doctoral scholarship.

I would like to extend my appreciation to Professor J. P. Martin Trusler from Imperial College and Dr. James Gunning from CSIRO, Australia, for agreeing to be my examiners.

I am grateful for the funding from the Qatar Carbonates and Carbon Storage Research Centre (QCCSRC), provided jointly by Qatar Petroleum, Shell, and Qatar Science & Technology Park. Many thanks also to Dr. Iain MacDonald and Professor Geoffrey Maitland.

Finally, I would like to thank my wife, Aline Leal, for her unlimited patience, kindness and love during these difficult years as a PhD student. Her company has made everything much easier! I would also like to thank my parents, Eli Mulin Leal and Benilda Moreira Leal, who are always supporting me and wishing me the best.

Contents

List of Figures

List of Tables

List of Symbols

Greek Symbols

α_i	the i-th species in the chemical system.
α_i^π	the i-th species in the π-th phase.
α_r	the convergence rate of an equilibrium calculation.
$\boldsymbol{\alpha}$	the set of species in the chemical system.
$\boldsymbol{\alpha}^\pi$	the set of species in the π-th phase.
$\bar{\xi}$	the projection constant in the stoichiometric equilibrium method.
ξ_i	the projection factor of the i-th Newton step in the stoichiometric equilibrium method.
$\boldsymbol{\varepsilon}$	the set of elements in the chemical system.
ϵ_Λ	the phase stability tolerance.
ϵ_{ill}	the threshold to determine when the Jacobian matrix is ill-conditioned.
ϵ_{tol}	the tolerance value used for the convergence checking.
ϵ_s	the stabilisation threshold in the stoichiometric equilibrium method.
ε_j	the j-th element in the chemical system.
φ_i	the fugacity coefficient of the gaseous species with index i.
γ_i	the activity coefficient of the aqueous species with index i.
Λ_π	the stability index of the π-th phase.
$\dot{\mu}$	the constant perturbation parameter used when the watchdog strategy is active.
$\hat{\mu}$	the perturbation parameter of the interior-point method.
μ_i	the chemical potential of the i-th chemical species.
μ_i°	the standard chemical potential of the i-th species.
μ_w	the threshold used to activate the watchdog strategy.
$\boldsymbol{\nu}$	the stoichiometric matrix of the linearly independent reactions in the chemical system.
ν_{ji}	the stoichiometry of the i-th species in the j-th reaction, and also the (i, j)-th entry of the stoichiometric matrix $\boldsymbol{\nu}$.
Ω_π	the generalised saturation index of the π-th phase.
Π	the number of phases in the chemical system.
θ_{ill}	the under-relaxation factor when the Jacobian matrix is ill-conditioned.

Roman Symbols

$\mathbf{a}$	the vector of activities of the chemical species.
a_{H^+}	the activity of the ionic species H^+.
a_i	the activity of the i-th chemical species.
$a_i^\star$	the specified activity of the i-th species.
$\mathbf{B}$	the balance matrix of the chemical system.

$\mathbf{b}$	the molar abundance vector of the elements.
b_j	the molar abundance of the j-th element.
$b_j^\star$	the specified number of moles of the j-th element.
$\mathbf{c}$	the equality constraint function in a optimisation problem.
C	the number of equilibrium constraints in the Gibbs energy minimisation problem.
$\mathbf{D}$	the diagonal scaling matrix of the primal variable $\mathbf{x}$.
$\mathbf{e}$	the vector of all ones.
$\mathbf{e}_i$	the vector whose i-th component is one, and all others zero.
E	the number of elements in the chemical system.
e	the optimality error function of the interior-point method.
$\mathbf{F}_{\hat{\mu}}$	the residual function of the perturbed KKT conditions.
f	the objective function in a optimisation problem.
G	the Gibbs free energy function of the chemical system.
$\mathbf{h}$	the vector-valued equilibrium constraint function.
$\mathcal{I}_\pi$	the set of indices of the species in the π-th phase.
$\mathcal{I}_g$	the indices of the gaseous species.
$i^\star$	the local index of the i-th species in its phase.
$\mathbf{K}$	the vector of equilibrium constants.
K_j	the equilibrium constant of the j-th reaction.
$\mathcal{L}$	the Lagrange function of a optimisation problem.
M	the number of linearly reactions in a chemical system.
m	the number of equality constraints in a optimisation problem.
m_i	the molality of the aqueous species with index i.
$\mathbf{n}$	the molar abundance vector of the chemical species.
N	the number of species in the chemical system.
n	the number of variables in a optimisation problem.
N_π	the number of species in the π-th phase.
n_i	the molar amount of the i-th chemical species.
$n_i^\star$	the specified number of moles of the i-th species.
n_t	the total molar amount in the chemical system.
n_t^π	the total molar amount in the π-th phase.
P	the pressure of the chemical system.
P°	the reference pressure for the activity of gaseous species.
P_i	the partial pressure of the i-th gaseous species.
R	the universal gas constant.
r_k	the calculation residual at the k-th iteration.
s_c, s_l	the scaling parameters in the convergence checking.
T	the temperature of the chemical system.
$\mathbf{W}$	the formula matrix of a chemical system.
$\mathbf{w}_j$	the j-th row of the formula matrix $\mathbf{W}$.
W	the number of watchdog iterations after which the algorithm checks for filter acceptance.
w_{ji}	the number of atoms of the j-th element in the i-th species, and also the (i, j)-th entry of the formula matrix $\mathbf{W}$.

$\bar{\mathbf{x}}$	the scaled primal variables.
$\hat{\mathbf{x}}$	the primal solution in a previous calculation.
$\mathbf{X}$	the diagonal matrix defined by $\mathbf{X} := \text{diag}(\mathbf{x})$.
$\mathbf{x}$	the vector of variables in a optimisation problem.
x_i^π	the molar fraction of the i-th species in the π-th phase.
x_i^g	the molar fraction of the i-th species in the gaseous phase.
$\mathbf{y}$, $\mathbf{z}$	the Lagrange multipliers of a optimisation problem.
y_j, z_i	the Lagrange multipliers with respect to the j-th constraint and i-th variable.
z	the vector of electrical charges of the species.

Mathematical Symbols

∇_x	the gradient operator with respect to the primal variables $\mathbf{x}$ only.
$\|\mathbf{x}\|_\infty$	the L^∞ norm of vector $\mathbf{x}$ defined as $\|\mathbf{x}\|_\infty := \max_i \|x_i\|$.
$\|\mathbf{x}\|_1$	the L^1 norm of vector $\mathbf{x}$ defined as $\|\mathbf{x}\|_1 := \sum_i \|x_i\|$.
$\|\mathbf{x}\|_2$	the L^2 norm of vector $\mathbf{x}$ defined as $\|\mathbf{x}\|_2 := \sqrt{\sum_i \|x_i\|^2}$.
^{T}	the transpose operator of a matrix or vector.

CHAPTER 1

INTRODUCTION

Geochemical modelling is of vital importance for the understanding of many environmental problems. Derived from concepts of chemical equilibrium and chemical kinetics, it allows the modelling of reactive processes that occur at the Earth's surface, subsurface or mantle. These include, for example, the complex interactions between natural waters, formation rocks, and gases, whose investigation is challenged by the large differences in the speed of the reactions, surface area effects, water salinity, temperature and pressure conditions, and others.

The current knowledge of geochemical modelling evolved from the first advances in chemical modelling. Interested in estimating the thrust of rocket fuels from its bulk composition, Huff et al. (1951) and Zeleznik and Gordon (1960) developed methodologies for equilibrium calculations for gaseous systems, which could then be used to determine the composition and volume of the combusted fuels, eliminating the need of time consuming and expensive tests.

Some years later similar approaches were applied to geochemical systems. The literature indicates that Garrels and Thompson (1962) were the pioneers in performing geochemical modelling. Their work consisted of hand computation of the species distribution in a sample of sea water at 25 °C and 1 bar from a measured bulk composition of the ionic species. This work was a breakthrough at the time, even though only simple thermodynamic models were considered.

Computational calculations did not take long to appear afterwards. Helgeson (1968) presented a computational method to calculate the reaction path of heterogeneous geochemical systems composed of an aqueous solution and minerals. This method consisted of integrating a system of differential equations parametrised not with a time variable, but with a reaction progress variable. It served as a basis for the subsequent works of Helgeson (1969) and Helgeson et al. (1969, 1970), and this research pioneered kinetic modelling of geochemical systems.

Many are the applications of geochemical modelling. It has been used extensively for speciation-solubility calculations using a variety of available packages in the literature (Truesdell and Jones, 1974; Nordstrom et al., 1979; Wolery, 1992b; Parkhurst and Appelo, 1999). It is very useful for managing radioactive waste disposal and understanding the contamination of groundwater, whose simulations are extremely important for informed political decisions and environmental policy development (Zhu and Anderson, 2002).

Another application is the modelling of carbon dioxide storage into geological formations as a measure to mitigate anthropogenic climate change (IPCC, 2005, 2013). For this, geochemical calculations and porous media flow simulations are essential to predict the fate of the injected carbon dioxide (Gunter et al., 1993, 1997; Kharaka et al., 2006). By accounting for the underground reactive processes between the greenhouse gas and the subsurface fluid and rock, it is possible to determine the amount of trapped carbon dioxide due to dissolution in the fluid and precipitation as secondary carbonate minerals.

Because of the mathematical complexity that emerges from thermodynamic and kinetics models in geochemical modelling, the use of advanced numerical methods is necessary. This work proposes, therefore, novel numerical methods for geochemical modelling and their computational implementation. The methods developed can model complex geochemical systems composed of many non-ideal phases, adopting chemical equilibrium or chemical kinetics models. These algorithms provide the core functionality for any geochemical calculation.

1.1 Chemical Equilibrium

In a chemical equilibrium state, the forward and reverse rates of the reactions in a system are equal, and therefore no changes in the concentrations of its species are observed with time. It is possible to demonstrate, with the use of the first and second laws of thermodynamics, that a chemical system undergoing an isobaric and isothermal process progresses towards a state of minimum Gibbs free energy. Other conditions also apply during this process, such as mass conservation of the chemical elements and charge balance condition of the aqueous solution.

Therefore, an equilibrium problem consists of finding the number of moles of the chemical species that simultaneously minimises the Gibbs free energy of the system and satisfies a system of equilibrium constraints (Smith and Missen, 1982). In addition, a non-negativity constraint for the number of moles is required in order to guarantee a physically feasible molar composition.

The applicability of chemical equilibrium solvers for environmental problems is wide. For instance, speciation modelling of aquatic systems, calculation of solubilities of gases and minerals, analysis of the effect of pH on the dissolution of a mineral, investigation of water-gas-rock effects during carbon storage in geological formations, and radioactive waste disposal modelling are all examples of problems that require equilibrium calculations.

In addition, it is a fundamental tool for chemical kinetics and subsurface reactive transport modelling. In these applications, some reactions in the system are controlled by thermodynamics instead of kinetics, and consequently equilibrium calculations are necessary (Lichtner, 1985; Steefel and Cappellen, 1990; Steefel and Lasaga, 1994; Steefel and MacQuarrie, 1996; Steefel et al., 2005).

Smith and Missen (1982) classify chemical equilibrium algorithms into two types: *stoichiometric* and *non-stoichiometric*. The former solves a system of mass-balance and mass-action equations, while the latter minimises the Gibbs free energy of the system. Zeleznik and Gordon (1968) and Van Zeggeren and Storey (1970) show, however, that both approaches are conceptually equivalent.

The literature contains many geochemical solvers for equilibrium calculations, most of them based on a stoichiometric approach. Examples of stoichiometric based solvers include WATEQ (Truesdell and Jones, 1974), MINEQL (Westall et al., 1976), WATEQ4F (Ball et al., 1987), MINTEQA2 (Allison and Kevin, 1991), EQ3/6 (Wolery, 1992a), MINEQL+ (Schecher and McAvoy, 1998), CHESS (van der Lee and Windt, 2002), PHREEQC (Parkhurst and Appelo, 1999, 2013), and The Geochemist's Workbench (Bethke, 2007).

In general, these solvers rely only on equilibrium constants of reactions, which are insufficient to calculate the Gibbs free energy of the geochemical system. As a result, determining the stable equilibrium phase assemblage is a difficult and expensive task. Many heuristic techniques have been developed to resolve this issue for pure mineral phases. For example, Bethke (2007) presents a technique to add the most supersaturated mineral to the calculation and remove the most undersaturated. Although popular among many geochemical solvers, this trial-and-error approach is inadequate and unsuitable for aqueous and gaseous phases. Consequently, many of these packages require that the aqueous and gaseous phase always exist at equilibrium, by fixing the amount of liquid water and the fugacity of a gaseous species.

Some issues are also found in the numerical methods of these stoichiometric geochemical solvers. For example, MINTEQA2 (Allison and Kevin, 1991), EQ3/6 (Wolery, 1992a), CHESS (van der Lee and Windt, 2002), PHREEQC (Parkhurst and Appelo, 1999, 2013), and The Geochemist's Workbench (Bethke, 2007) adopt an incomplete Newton scheme developed by Morel and Morgan (1972) and further improved by Reed (1982) for solving aqueous speciation. The approach consists of arranging the species in a set of primary species and another of secondary species. The composition of the primary species is calculated by applying Newton's method to the modified mass-balance equations. The composition of the secondary species, on the other hand, is calculated via a successive substitution approach using the mass-action equations.

This incomplete Newton scheme aims to reduce the dimension of the Jacobian matrix so that less computational effort is spent in solving linear systems. However, combining Newton's method with a successive substitution approach has the potential of preventing the calculation to converge at quadratic rates near the solution. Moreover, in some chemical equilibrium calculations, the cost of evaluating expensive equations of state and their derivatives could exceed the cost of solving linear systems. See Appendix A for a more detailed description of this stoichiometric formulation and an in-depth discussion on its convergence rate.

Another typical approach in these geochemical solvers is the assumption of specific details about species, reactions, and phases in the equilibrium calculation. These details can be, for example, an arbitrary decision to neglect the reactive consumption/production of water (Allison and Kevin, 1991), the assumption that an aqueous phase always exists (Allison and Kevin, 1991; Bethke, 2007; Parkhurst and Appelo, 1999, 2013), or the assumption that a gaseous phase exists only under the imposition of a specified partial pressure or fugacity of a gaseous species (Wolery and Daveler, 1992; Wolery, 1992a,b; Bethke, 2007).

The disadvantages that follow from this approach are many. Firstly, handling so many particular and arbitrary cases can result in code bloat, which reduces efficiency and maintainability of the software. Secondly, it becomes complicated to add new features to the solver, since the introduction of new thermodynamic models and phase types might incur several modifications throughout the code. Finally, restricting specific thermodynamic models for some types

of species restricts the solver flexibility and applicability, preventing it from performing customised computations using the plethora of models existent in the literature.

Geochemical packages based on Gibbs energy minimisation algorithms include ChemSage (Eriksson and Hack, 1990), THERIAC (de Capitani and Brown, 1987), HCh (Shvarov, 1999, 2008), FactSage (Bale et al., 2002, 2009), and GEM-Selektor (Karpov et al., 1997, 2001, 2002; Kulik et al., 2004, 2013).

In ChemSage (Eriksson and Hack, 1990), a Gibbs energy minimisation algorithm is implemented in which the non-ideal values of the chemical potentials are used, but their derivatives correspond to ideal models. Although convergence might not be an issue with such practice (e.g., quasi-Newton methods converge with artificial derivatives that are considerably different from their real values, Nocedal and Wright, 1999; Fletcher, 2000), the approach cannot converge at a quadratic rate near the solution. In Eriksson and Hack (1990) it is pointed out that a strategy was developed to refine the initial guess to improve the efficiency of the algorithm. However, this improvement cannot change the convergence rate of the method.

In GEM-Selektor (Karpov et al., 1997, 2001, 2002; Kulik et al., 2004; Wagner et al., 2012; Kulik et al., 2013), an interior-point algorithm is used to minimise the Gibbs free energy of multiphase systems. Their method, however, does not use the logarithmic barrier functions of Fiacco and McCormick (1990) nor the KKT perturbation approach of El-Bakry et al. (1996), which are practices commonly adopted in several non-linear programming packages and algorithms (Ulbrich et al., 2004; Silva et al., 2008; Wächter and Biegler, 2005a,b,c; Vanderbei, 1999; Benson et al., 2000; Vanderbei, 2006; Byrd et al., 1999; Costa and Fernandes, 2008, 2011a,b). GEM-Selektor assumes that at every iteration the minimisation of the Gibbs energy is a convex problem, and uses this assumption to compute a descent direction. This direction is then used to find a step length that sufficiently minimises the Gibbs free energy, a procedure similar to a line-search strategy. The method not only maintains feasibility on the bounds at every iteration, but also on the mass-balance constraints.

Harvie et al. (1987) presents an equilibrium algorithm for non-ideal multiphase systems based on the minimisation of the Gibbs free energy. Their algorithm consists of transforming a constrained minimisation problem into an unconstrained one by introducing Lagrange multipliers and quadratic slack variables to circumvent the bound constraints. Moreover, their approach has some similarity to active-set minimisation methods, since their calculated directions are always feasible and some species are sometimes made active (i.e., having zero number of moles) during the search for the stable phase assemblage. However, their method presents some strategies for finding the global minimum, an approach not attempted in this work, since its computational cost would make the proposed method prohibitive for reactive transport simulations.

In Perez et al. (2012), a method for multiphase geochemical speciation calculation is presented. The numerical method consists of combining two-phase flash calculations[1] with aqueous speciation, where two independent routines that communicate with each other are used to solve the multiphase equilibrium problem. This differs from the methodology presented in this work,

[1] An approach for multiphase chemical equilibrium calculations in which only a simple one-to-one relationship between the components can exist, differently from the general approach based on the law of mass-action presented later in Chapter 2.

which solves the full equilibrium problem without making separate flash calculations to account for a gaseous phase.

In the recent work of Harvey et al. (2013), a Gibbs free energy minimisation method for systems composed of solid solutions was developed. Their method uses exact Hessian expressions, which potentially produce fast convergence rates near the solution. They compared their method with the general-purpose optimisation packages IPOPT (Wächter and Biegler, 2005c), KNITRO (Byrd et al., 1999), and SNOPT (Gill et al., 2002, 2005), identifying the necessity of suitable heuristics to improve the efficiency and robustness of chemical equilibrium calculations. Moreover, they observed that using quasi-Newton approximations for the Hessian matrix, an approach adopted by the general optimisation solver SNOPT, results in severely inefficient chemical equilibrium calculations. In their algorithm, all phases and species are initially assumed to exist at equilibrium, allowing a simplification of the KKT equations by eliminating the complementary equations. However, some phases can be excluded during the calculation if they are detected as unstable. The authors define an unstable phase as a phase that for a certain number of past iterations a full Newton step would bring its molar abundance to a negative value. These phases are then removed because eliminating the complementary conditions from the KKT equations is subjected to all species having non-zero number of moles. In the end of the calculation, phase stability tests are performed to identify any excluded phase that should be added to the system, or any phase at equilibrium that should be removed. Our algorithm, on the other hand, does not exclude or add any phase in the course of the calculation to attain numerical stability. All phases are assumed in the calculation, and the unstable phases are identified in the end of the calculation with a rigorous phase stability test.

1.2 Chemical Kinetics

Frequently geochemical investigations of a system assume chemical equilibrium conditions. Calculating the solubilities of minerals and gases in aqueous solutions at different temperatures, pressures, salinities, and other conditions require only equilibrium calculations (Anderson and Crerar, 1993). Sometimes, however, one might be interested in the time scales over which such processes occur, and equilibrium calculations cannot provide this.

Application of chemical kinetics theory is vital when the transient chemical state of a system is important. This is useful, for example, to analyse the temperature and pressure effects on the time required for a mineral to equilibrate with a solution. In addition, it describes the transient water-gas-rock effects over time of a geochemical process, such as the continuous consumption or production of gases while minerals are reacting in an aqueous solution.

A more detailed modelling procedure has its consequent complexities, however. Chemical kinetics calculations require more input data and models than chemical equilibrium calculations. For example, calculating the evolution of the system composition demands rate laws of the reactions. In addition, due to its time-dependence, chemical kinetics consists of solving a system of ordinary differential equations, while chemical equilibrium requires only the solution of algebraic equations.

Another complexity in geochemical kinetics is the broad difference of the speeds of the aqueous, gaseous and mineral reactions (Lasaga, 1998). Commonly, aqueous reactions proceed sub-

stantially faster than mineral reactions, with the former sometimes achieving equilibrium in microseconds, while the latter requiring several days to many years (Zhang, 2008). Therefore, this can result in an inefficient numerical integration of the ordinary differential equations, which would need tiny time steps to ensure accuracy and stability.

To address this problem, the geochemical system is considered to be in *partial equilibrium* (Helgeson, 1968; Helgeson et al., 1969, 1970). A system in partial equilibrium means that it is in equilibrium with respect to some reactions and out of equilibrium with respect to others. For example, since aqueous and gaseous reactions are often considerably faster than mineral reactions, it seems plausible to assume they are in equilibrium at all times. As the mineral reactions proceed kinetically, the aqueous and gaseous reactions are constantly perturbed and then instantaneously re-equilibrated. Note, however, that the partial equilibrium assumption is based on the relative speed of the reactions. Therefore, it is possible to assume a mineral reaction in equilibrium and an aqueous reaction out of equilibrium at any instant.

The partial equilibrium assumption simplifies the problem by replacing stiff differential equations by algebraic ones. These algebraic equations govern the equilibrium condition of those reactions assumed to be in equilibrium. As a result, the governing equations become a system of non-linear differential-algebraic equations, with the constraint that mass of the chemical elements in the system must be conserved.

The first work on computational reaction path modelling in geochemistry can be tracked to Helgeson (1968) and Helgeson et al. (1969). They presented a modelling example of the hydrolysis of K-feldspar, where the partial equilibrium assumption was adopted by considering the aqueous reactions in equilibrium. However, their reaction path modelling was not based on kinetic rate laws of the reactions, but on the use of a progress variable to describe the compositional change of the system. Helgeson (1971) later on modelled the feldspar hydrolysis using a parabolic rate law. The simplistic rate was derived using Fick's first law of diffusion on a one-dimensional problem of diffusion along the surface layer of the mineral reacting with the aqueous solution.

Helgeson and Murphy (1983) combined the rate laws proposed in Aagaard and Helgeson (1982) with a numerical integration routine to model irreversible reactions among minerals and aqueous solutions. They again considered the hydrolysis of feldspar, with the possibility of precipitation of secondary minerals (muscovite, gibbsite, and kaolinite). The secondary minerals were assumed to precipitate under the partial equilibrium assumption.

Although these preliminary works led by Helgeson 30–40 years ago were the precursors of many others, they were always intended for a specific system. There was no formalisation of geochemical kinetics calculations for general multiphase systems with a mixing of equilibrium- and kinetically-controlled reactions. In addition, no efficient methodology was discussed for the solution of the resulting system of differential-algebraic equations. Some of these gaps have been addressed, as discussed next. However, an efficient, general, and flexible algorithm has yet to be developed for chemical kinetics in geochemical modelling.

The following is a list of computer codes commonly used for geochemical kinetic modelling: EQ6 (Wolery and Daveler, 1992), PHREEQC (Parkhurst and Appelo, 1999, 2013), MINTEQA2 (Allison and Kevin, 1991), CHESS (van der Lee and Windt, 2002), SOLMINEQ.88 (Kharaka et al., 1988), and The Geochemist's Workbench (Bethke, 2007). They calculate the evolution

of systems as minerals kinetically dissolve or precipitate. In addition The Geochemist's Workbench, as described in Bethke (2007), provides support for modelling redox reactions controlled by kinetics.

As discussed in Section 1.1, these geochemical packages adopt a stoichiometric approach for aqueous speciation calculations. Their databases contain only the equilibrium constants of the reactions, which are required for the solution of the system of mass action equations. The chemical potentials of the species, on the other hand, are not available, which are necessary to calculate the Gibbs free energy of the system. Therefore, determining the stable equilibrium phase assemblage of the system is a difficult task, since, given two or more states, it is not possible to determine which one has the lowest Gibbs free energy. In addition, these geochemical codes use an incomplete Newton scheme for aqueous speciation calculations, which results in slow rates of convergence, as discussed in Appendix A.

Recently, Mironenko and Zolotov (2011) developed a computer code for modelling equilibrium-kinetics of water-rock interactions. Instead of using a stoichiometric scheme for chemical equilibrium calculations, they used the algorithm of de Capitani and Brown (1987), which minimises the Gibbs free energy of the system using a simplex approach. The chemical equilibrium algorithm used here, however, is capable of minimising non-convex objective functions using a trust-region interior-point method. In addition, the adaptive time step control adopted in this work is based on numerical analysis, while their approach is based on a heuristic that aims to prevent strong changes in pH between time steps.

1.3 Carbon Capture and Storage

Global warming has become an important societal concern. Anthropogenic emission of greenhouse gases to the atmosphere has been detected as the primary cause for current climate changes (Staudt et al., 2008). Unprecedented energy production from burning fossil fuels such as oil, coal and natural gas has substantially elevated the concentration of CO_2 in the atmosphere over the past few decades (Houghton et al., 2001).

In order to minimize anthropogenic emissions of CO_2 to the atmosphere and mitigate climate changes, the technique known as Carbon Capture and Storage (CCS) was developed. The approach consists of the capture of CO_2 from static sources (e.g., fossil fuel power plants) and its storage into geological formations (e.g., deep saline aquifers, depleted oil/gas reservoirs). According to Ormerod (1994), the injection of supercritical CO_2 into deep saline aquifers is one of the most promising storage process due to the highest storage capacity of these formations.

As supercritical CO_2 is injected into a deep saline aquifer, four main trapping mechanisms can occur, which will promote the isolation of carbon dioxide from the atmosphere. These are listed below:

Structural trapping is the trapping mechanism promoted by an impermeable layer of cap-rock that prevents the injected supercritical CO_2 to escape to the atmosphere. In the absence of this cap-rock, the supercritical carbon dioxide can potentially leak due to buoyancy effects, since its density is less than that of the subsurface fluid. This is the major trapping mechanism in the short term of a carbon storage process, as illustrated in Figure 1.1,

since at the initial stages of injection most CO_2 is mobile (IPCC, 2005). Its importance is gradually diminished as the mobility of the greenhouse gas decreases with its continue dissolution into the subsurface fluid, its reaction with the porous rock, and its trapping by capillary effects.

Residual trapping is the trapping mechanism that consists of the immobilization of supercritical CO_2 by capillary forces. Because of this, it is also known as *capillary trapping*. As the plume of supercritical carbon dioxide migrates through the porous media, small bubbles of the gas is gradually trapped inside the porous rock. This trapping occurs because of the snap-off phenomenon that consists of brine flowing through the porous without displacing those small CO_2 bubbles. As a consequence, a considerable amount of carbon dioxide remains immobile in the saline aquifer, and, therefore, safely isolated from the atmosphere. Recent measurements of residual trapping of supercritical CO_2 in both sandstones and carbonate rocks can be found in Pentland et al. (2011) and El-Maghraby and Blunt (2013).

Solubility trapping consists of the trapping of the injected supercritical carbon dioxide as dissolved gas in the subsurface fluid. As the CO_2 plume spreads throughout the aquifer, it continually dissolves in the resident brine. The amount of dissolution depends on pressure, temperature, brine salinity, and the extent of the contact area between the CO_2 plume and brine. Further dissolution of the mobile gas occurs because of the gravity-driven convective mixing that takes place (Lindeberg and Wessel-Berg, 1997). In this physical process, the denser brine saturated with carbon dioxide sinks and the unsaturated brine rises, further dissolving the available mobile supercritical carbon dioxide. The numerical investigations of Ennis-King and Paterson (2005) estimated the start time of the convective mixing process for many permeability settings, which varied from one year to hundreds of years after carbon dioxide injection.

Mineral trapping consists of immobilising the injected CO_2 as precipitates of secondary carbonate minerals. Since the dissolution of carbon dioxide in brine forms an acidic solution, geochemical reactions with the primary minerals in the host porous rock can lead to the formation of secondary carbonate minerals over time, indirectly sequestering carbon. Because of the slow reaction rates, however, this trapping mechanism becomes important only over larger time scales (hundreds to thousands of years depending on the composition of the rock minerals). This is a secure trapping mechanism, since CO_2 will remain trapped away from the atmosphere for geological times (Gunter et al., 1993).

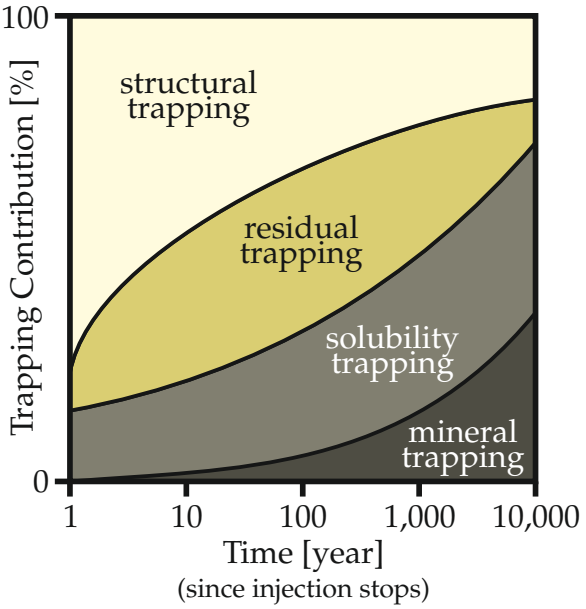
The simultaneous quantitative estimation of all these trapping mechanisms is a daunting task. A complete and accurate analysis of the ultimate fate of carbon dioxide after its injection depends on the interplay of several physical and chemical processes. These processes in turn are strongly coupled and, as a consequence, an individual and independent investigation of a single trapping mechanism may not reflect the true immobilization of CO_2.

A comprehensive study of the long-term storage of CO_2 in geological formation by means of laboratory experimentation is impractical for several reasons. These include, for example, the difficulties in representing the ambient conditions deep underground and the slow physical

and chemical processes in the porous media flow, such as geochemical reactions that can take hundreds to thousands of years to equilibrate. Therefore, the use of reactive flow simulations for the fate assessment of the injected carbon dioxide is fundamental.

In order to account for the water-gas-rock effects that occur during storage of carbon dioxide in deep saline aquifers, geochemical reactions must be considered (Gunter et al., 1993; Gaus et al., 2008). This is important, for example, to determine the overall trapping of the greenhouse gas by both solubility and mineral mechanisms. In addition, accounting for the heterogeneous reactions between fluid and rock has also considerable impact on the calculated amount of dissolved carbon dioxide and the pH of the brine (Kharaka et al., 2006).

Figure 1.1: Contribution of the four main trapping mechanisms in carbon dioxide sequestration in saline aquifers (IPCC, 2005).



1.4 Objectives

The numerical solution of the porous media flow equations are very well understood. Advances in numerical methods for partial differential equations and adaptive mesh refinement schemes in the past few decades have allowed several physical processes such as advection, diffusion and dispersion to be represented fairly accurately. However, numerical methods for modelling chemical processes in the subsurface flow have yet much to be improved.

Improvement in these methods are needed for various reasons. As discussed in Sections 1.1 and 1.2, most algorithms for chemical equilibrium and kinetics calculations were developed in the past for standalone geochemical calculations. Therefore, their immediate applicability into critical applications such as reactive transport simulators, which requires thousands to millions of equilibrium and kinetics calculations every time step, might not be optimal.

Take as an example the recently updated geochemical package GEM-Selektor (Kulik et al., 2013; Wagner et al., 2012), which was not initially developed for integration into reactive transport simulations. Its new equilibrium algorithm GEMS3K is an improvement over the previous GEMIPM2K, whose accuracy and stability were not sufficient for reactive transport modelling (Shao et al., 2009). Therefore, this need of Kulik et al. (2013) of adapting their numerical method for critical geochemical applications raises the question of whether other approaches in the geochemical literature require some adaptations as well. Since many geochemical equilibrium methods were initially developed with the primary intent for speciation-solubility calculations and plotting phase diagrams, this might indeed be the case.

Reactive transport simulators very often explicitly combine both transport equations and chemical equations. Consequently, this results in complicated and rigid mathematical and computational formulations that becomes difficult to add any new feature, such as support for new phases and reactions. Moreover, the chemical processes in the subsurface flow cannot be calculated by dedicated numerical methods and specific computer routines for equilibrium and kinetics problems. Examples of simulators that adopt this practice are TOUGHREACT (Xu et al., 2004, 2006), PFLOTRAN (Lu and Lichtner, 2005), and CrunchFlow (Steefel, 2009). Therefore, this class of reactive transport simulators can benefit from the development of efficient, robust and accurate equilibrium and kinetics methods.

This work aims to develop modern numerical methods for geochemical processes that are at the same time widely applicable and highly efficient for reactive transport simulators. The mathematical formulation of the methods assumes no specific details about the chemical system. Therefore, their applicability is broad, allowing them to be used with any combination of phases, species, and thermodynamic models.

Chapter 2 presents a numerical method for chemical equilibrium calculations based on the stoichiometric approach. Unlike the traditional approach of imposing only mass-balance and charge-balance conditions on the chemical elements of the system, the presented method permits the specification of other types of equilibrium constraints such as fixed activity or fugacity of a species, fixed partial pressure of a gas, fixed concentration or amount of a species, and others. This generality in specifying equilibrium constraints is important, for example, if the solubility of a mineral needs to be calculated over a range of pH values.

Chapter 3 presents a chemical equilibrium method based on a non-stoichiometric approach. The method consists of directly minimising the Gibbs free energy of the system using a primal-dual interior-point method. The advantage of this equilibrium algorithm over the previous one is that determining the unstable phases (e.g., phases that should not be at equilibrium) is more efficient and robust. In addition, this method uses novel numerical methods for non-convex optimisation problems in the mathematical literature.

Chapter 4 presents an algorithm for chemical kinetics calculation in multiphase geochemical systems. The method supports the mixing of both equilibrium- and kinetically-controlled reactions. The necessary chemical equilibrium calculations are performed using the Gibbs free energy minimisation algorithm presented in Chapter 3. Integration of the ordinary differential equations is performed using the implicit multistep BDF algorithm (Ascher and Petzold, 1998). An adaptive control scheme of the time step is adopted for efficient integration of the differential equations.

Chapter 5 presents results using the chemical equilibrium methods of Chapters 2 and 3 to calculate multiphase equilibrium calculations for problems related to CO_2 sequestration in saline aquifers. These include the numerical investigation of the solubility of carbon dioxide in a variety of brines, the pH of brines saturated with CO_2, the solubility of carbonate minerals in the presence of carbon dioxide, and the calculation of phase behaviour in systems composed of aqueous, gaseous and mineral phases.

Chapter 6 presents results using the chemical kinetics method of Chapter 4 to simulate the kinetic water-gas-rock effects during CO_2 injection in saline aquifers. The kinetic modelling aims to analyse the fate of the injected supercritical CO_2 on both carbonate and sandstone saline

aquifers, determining the extent at which this greenhouse gas is trapped by solubility and mineral mechanisms. A kinetic modelling of the dissolution of pure calcite in CO_2 saturated water is also performed, and the calculations compared with experimental measurements.

Chapter 7 summarises this work and presents conclusions and recommendations for future work.

CHAPTER 2

CHEMICAL EQUILIBRIUM: STOICHIOMETRIC APPROACH

In this chapter a chemical equilibrium algorithm based on a stoichiometric approach is presented. The methodology is developed for general multiphase systems, where no presumption on the types of phases, species and reactions is made. The method adopts a modified Newton's method to solve a system of mass-action equations coupled with a system of equilibrium constraints. A stabilisation procedure is developed to promote convergence of the calculation when a phase in the chemical system is absent in the equilibrium state.

2.1 Introduction

As introduced in Section 1.1, an equilibrium problem is a constrained minimisation problem, where the Gibbs free energy of the system is at a minimum at equilibrium. However, approaches for equilibrium calculations exist that do not formally minimise the Gibbs free energy of the system. These replace the minimisation problem by a system of non-linear equations known as *mass-action equations*. These equations, which are derived from the first-order optimality conditions of the Gibbs energy minimisation problem (Leal et al., 2013), model the equilibrium condition of the reactions, that is, their equal forward and reverse reaction rates.

In what follows, assume a chemical system composed of N species, where the i-th species is denoted by α_i and the set of species by $\boldsymbol{\alpha} = \{\alpha_1, \ldots, \alpha_N\}$. In addition, consider that there exist E elements from which these species can be formed, where the j-th element is denoted by ε_j and the set of elements by $\boldsymbol{\varepsilon} = \{\varepsilon_1, \ldots, \varepsilon_E\}$. Finally, assume that the chemical species are partitioned among Π phases, where α_i^π denotes the i-th species in the π-th phase, $\boldsymbol{\alpha}^\pi = \{\alpha_1^\pi, \ldots, \alpha_{N_\pi}^\pi\}$ denotes the set of species in the π-th phase, N_π denotes the number of species in the π-th phase, and $\mathcal{I}_\pi$ denotes the set of indices of the species in the π-th phase.

The chemical equilibrium algorithm based on the stoichiometric approach is formulated from a system of linearly independent reactions, which can be determined from the given set of

species $\boldsymbol{\alpha}$. The number of these reactions is here denoted by M. In what follows we will see how these linearly independent reactions are obtained.

Definition 2.1. Let w_{ji} denote the number of atoms of the j-th element in the i-th species. The *formula matrix* of the system is defined as the matrix $\mathbf{W} \in \mathbb{R}^{E\times N}$ whose (j,i)-th entry is w_{ji}.

Definition 2.2. Let $\boldsymbol{z} \in \mathbb{R}^N$ denote the vector of electrical charges of all species. Define the *balance matrix* $\mathbf{B} \in \mathbb{R}^{(E+1)\times N}$ as:

$$\mathbf{B} = \begin{bmatrix} \mathbf{W} \\ \boldsymbol{z}^T \end{bmatrix}. \tag{2.1}$$

Definition 2.3. Let ν_{ji} denote the stoichiometry of the i-th species in the j-th reaction, where ν_{ji} is positive for products and negative for reactants. Define the *stoichiometric matrix* of the reactions as the matrix $\boldsymbol{\nu} \in \mathbb{R}^{M\times N}$ whose (j,i)-th entry is ν_{ji}.

As a result of the above definition, the reactions taking place in the system can be written in general form as:

$$0 \rightleftharpoons \sum_{i=1}^{N} \nu_{ji}\alpha_i \qquad (j = 1,\ldots,M). \tag{2.2}$$

A well-defined reaction is both mass and charge balanced. In other words, the number of atoms of each chemical element and the electrical charge, on both sides of the reaction, must be equal. Therefore, if $\boldsymbol{\nu}_j$ denotes the j-th row of matrix $\boldsymbol{\nu}$, which contains the stoichiometric coefficients of the j-th reaction, it follows that:

$$\mathbf{B}\boldsymbol{\nu}_j^T = \mathbf{0}. \tag{2.3}$$

Hence, any reaction in the system must satisfy:

$$\boldsymbol{\nu}_j^T \in \ker(\mathbf{B}), \tag{2.4}$$

where $\ker(\mathbf{B}) = \{\mathbf{x} \in \mathbb{R}^N : \mathbf{B}\mathbf{x} = \mathbf{0}\}$ is the kernel of the balance matrix $\mathbf{B}$.

Condition (2.4) forms the basis for the determination of a set of linearly independent reactions involving the species $\boldsymbol{\alpha}$. It tells that the matrix $\boldsymbol{\nu}^T$ is a kernel of the balance matrix $\mathbf{B}$, and so its largest number of linearly independent columns $\boldsymbol{\nu}_j^T$, (i.e., the number of reactions M) is given by $M = \text{nullity}(\mathbf{B})$. This can be further simplified to:

$$M = N - C,$$

where $C = \text{rank}(\mathbf{B})$.

Example 2.1. *Determining the reactions in the system H_2O–CO_2–NaCl*

Consider a system composed by the following chemical species:

$$\boldsymbol{\alpha} = \{\text{H}_2\text{O(l)},\ \text{H}^+,\ \text{OH}^-,\ \text{Na}^+,\ \text{Cl}^-,\ \text{NaCl(aq)},\ \text{HCO}_3^-,\ \text{CO}_3^{2-},\ \text{CO}_2\text{(aq)}\}, \tag{2.5}$$

whose chemical elements are:

$$\boldsymbol{\varepsilon} = \{\text{H},\ \text{O},\ \text{Na},\ \text{Cl},\ \text{C}\}. \tag{2.6}$$

The formula matrix of this system is:

$$\mathbf{W} = \begin{array}{c} \\ \text{H} \\ \text{O} \\ \text{Na} \\ \text{Cl} \\ \text{C} \end{array} \begin{array}{c} \begin{array}{ccccccccc} \text{H}_2\text{O(l)} & \text{H}^+ & \text{OH}^- & \text{Na}^+ & \text{Cl}^- & \text{NaCl(aq)} & \text{HCO}_3^- & \text{CO}_3^{2-} & \text{CO}_2\text{(aq)} \end{array} \\ \left[\begin{array}{ccccccccc} 2 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 3 & 3 & 2 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{array}\right] \end{array}. \tag{2.7}$$

It can be shown in this particular case that the balance matrix $\mathbf{B} = \mathbf{W}$, since the rows of the formula matrix $\mathbf{W}$ are linearly independent and the vector of charges $\boldsymbol{z}$ is a linear combination of the rows of $\mathbf{W}$.

Once the balance matrix $\mathbf{B}$ is obtained, one can then calculate its kernel, and obtain the following stoichiometric matrix by transposing the result:

$$\boldsymbol{\nu} = \begin{bmatrix} -1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & -1 \end{bmatrix}, \tag{2.8}$$

which defines the following system of linearly independent reactions:

$$\text{H}_2\text{O(l)} \rightleftharpoons \text{H}^+ + \text{OH}^-, \tag{2.9}$$

$$\text{NaCl(aq)} \rightleftharpoons \text{Na}^+ + \text{Cl}^-, \tag{2.10}$$

$$\text{HCO}_3^- \rightleftharpoons \text{CO}_3^{-2} + \text{H}^+, \tag{2.11}$$

$$\text{CO}_2\text{(aq)} + \text{H}_2\text{O(l)} \rightleftharpoons \text{HCO}_3^- + \text{H}^+. \tag{2.12}$$

2.2 Governing Equations

Assume a multiphase system whose chemical species participate in the following linearly independent reactions:

$$0 \rightleftharpoons \sum_{i=1}^{N} \nu_{ji} \alpha_i \qquad (j = 1, \ldots, M). \tag{2.13}$$

Since the system is assumed to be in equilibrium, the composition of the chemical species must satisfy the following *mass-action equations* (Smith and Missen, 1982):

$$K_j(T, P) = \prod_{i=1}^{N} a_i(T, P, \mathbf{n})^{\nu_{ji}} \qquad (j = 1, \ldots, M), \tag{2.14}$$

where $K_j \colon \mathbb{R}^2 \mapsto \mathbb{R}$ denotes the equilibrium constant function of the j-th reaction; $a_i \colon \mathbb{R}^{2+N} \mapsto \mathbb{R}$ the activity function of the i-th chemical species; T and P the temperature and pressure of the system respectively; $\mathbf{n} \in \mathbb{R}^N$ the molar abundance vector of the system, whose i-th component n_i denotes the molar amount of the i-th species.

In the literature, the activity of a species is commonly replaced by the product of a concentration quantity and an activity coefficient. For example, the activity of an aqueous species is replaced by:

$$a_i = m_i \gamma_i, \tag{2.15}$$

where m_i and γ_i are its molality and activity coefficient. The activity of a gaseous species, on the other hand, is replaced by:

$$a_i = \varphi_i x_i^g \frac{P}{P^\circ}, \tag{2.16}$$

where x_i^g and φ_i are its molar fraction and fugacity coefficient, and P° is a reference pressure, usually equal to 1 bar, though this depends on the standard chemical potential data of the gaseous species.

In our formulation, however, the term a_i, representing the activity of a species, is included in all equations. The idea is to avoid the explicit dependence of the type of the species every time an equation is written. For example, the mass-action equations (2.14), and also the Gibbs energy function (3.2) later in Chapter 3, can be succinctly and generically written without the need to write terms that corresponds to aqueous, gaseous, mineral, and possibly other types of species. This is a fundamental step for the development of a general chemical equilibrium algorithm that can be applied to any type of systems. At the computational level, such abstraction is also possible to be made with programming techniques such as polymorphism.

The equilibrium constants K_j of the reactions are usually interpolated at a given temperature and pressure from known values. These values can be either obtained experimentally, or evaluated using:

$$\ln K_j(T, P) \coloneqq -\frac{1}{RT} \sum_{i=1}^{N} \nu_{ij} \mu_i^\circ(T, P), \tag{2.17}$$

where $\mu_i^\circ \colon \mathbb{R}^2 \mapsto \mathbb{R}$ is the standard chemical potential of the i-th species; and R is the universal gas constant (Smith and Missen, 1982).

The interpolation approach is an efficient and accurate approach. It is justified by the fact that the evaluation of the equations of state to compute the standard chemical potentials μ_i° can be prohibitively expensive. However, the same practice cannot be done with the activities a_i, because these functions depend on the composition of the system. Therefore, they must be directly computed from their equations of state. Moreover, fast convergence of the equilibrium calculations requires at least their first-order partial molar derivatives.

The number of mass-action equations (2.14) is not enough to resolve the equilibrium state of the system. Assuming that temperature and pressure are specified, the problem contains a total of N unknowns, which are the molar amounts n_i of the species. Since there are M mass-action equations, an additional of $C = N - M$ equilibrium constraints are required to resolve the problem.

Recall, however, that the mass of the elements are conserved during the equilibration process and the electroneutrality condition must be attained. Therefore, let $\mathbf{b} \in \mathbb{R}^E$ denote the molar abundance vector of the elements, whose j-th component b_j is the molar amount of the j-th element in the system. It follows that:

$$\mathbf{W}\mathbf{n} = \mathbf{b} \tag{2.18}$$

and

$$\boldsymbol{z}^T \mathbf{n} = \mathbf{0} \tag{2.19}$$

are the missing constraints to resolve the equilibrium problem, known as the *mass-balance* and *charge-balance equations*. Note that there are $E + 1$ equations in (2.18) and (2.19). Depending

on the chemical system, some of these equations can be linearly dependent. It can be shown that after removing the linearly dependent equations, a total of $C = \text{rank}(\mathbf{B})$ equilibrium constraints is obtained.

The mass-balance and charge-balance equations are the natural constraints to be imposed for chemical equilibrium calculations. In geochemical modelling, however, very frequently the modeller is interested in imposing other kinds of constraints. Examples include the pH of the solution, the total concentration of ionic species, and the electro-neutrality condition of the mixture (Anderson and Crerar, 1993; Bethke, 2007). Below is a brief list of the constraints that can be imposed at equilibrium:

- the number of moles of an element;
- the number of moles of a species;
- the charge-balance condition of the aqueous solution;
- the activity of a species; and
- the partial pressure of a gaseous species.

Therefore, the equilibrium constraints in this work is represented by the general equation:

$$\mathbf{h}(\mathbf{n}; T, P) = \mathbf{0}, \tag{2.20}$$

where $\mathbf{h}\colon \mathbb{R}^{N+2} \mapsto \mathbb{R}^{E}$ is the equilibrium constraint function assumed to be continuously differentiable. For simplicity, this dependence on temperature and pressure will be omitted.

Observe that this approach for specifying equilibrium constraints is general and flexible. For example, the mass-balance and charge-balance equations (2.18) and (2.19) can be defined as:

$$\mathbf{h}(\mathbf{n}) := \begin{bmatrix} \mathbf{W}\mathbf{n} - \mathbf{b} \\ z^T\mathbf{n} \end{bmatrix}. \tag{2.21}$$

and its gradient as:

$$\nabla\mathbf{h}(\mathbf{n}) := \mathbf{B}. \tag{2.22}$$

Let us now show how some other equilibrium constraints can be constructed individually. These individual constraints will be denoted by the function $h\colon \mathbb{R}^N \mapsto \mathbb{R}$, which can be combined to form the vector-valued function $\mathbf{h}$.

Imposition of the number of moles of an element

There are equilibrium problems in which the number of moles of some elements are unknown. Instead of specifying these amounts, they are in fact calculated, which constitutes an inverse problem (Kulik, 2006; Kulik et al., 2013).

As an example, consider the case where one needs to find out the amount of hydrochloric acid that needs to be mixed with 1 kg of water to produce a solution with pH equal to 4.0. In this problem, therefore, the number of moles of elements H and Cl are unknown a priori. However, these amounts can be determined by solving an equilibrium problem where *(i)* the pH of the solution is imposed together with *(ii)* the condition of charge-balance of the aqueous solution

and *(iii)* the known amount of element oxygen. Thus, we see that the capability of specifying the number of moles of individual elements is of fundamental importance for chemical equilibrium modelling.

Imposing the number of moles of the j-th element is governed by the following equilibrium constraint function:

$$h(\mathbf{n}) := \mathbf{w}_j^T \mathbf{n} - b_j^\star, \tag{2.23}$$

where $b_j^\star$ denotes the desired number of moles of the j-th element; and $\mathbf{w}_j \in \mathbb{R}^N$ the vector formed from the j-th row of the formula matrix $\mathbf{W}$. The gradient of this function is given by:

$$\nabla h(\mathbf{n}) := \mathbf{w}_j. \tag{2.24}$$

Imposition of the number of moles of a species

Frequently, equilibrium problems are formulated from a mixing of a chemical species. For example, a mixing of 1 kg of water with 100 g of supercritical CO_2. Then, from these given amounts, the number of moles of the chemical elements can be readily determined and a forward equilibrium calculation performed.

There are cases, however, in which one might be interested in the effect that the amount of a given species has on the equilibrium state of the system. For example, at a given temperature and pressure, how much supercritical CO_2 must be mixed with 1 kg of water in a closed system to produce 1 g of vapour water? In this inverse problem, we must first identify the elements whose amounts are unknown in the problem, which are C and O. Thus, the solution of this problem can be easily obtained by calculating an equilibrium problem whose constraints are *(i)* the mass of species H_2O(g) is 1 g, *(ii)* the molar amount of H is twice the number of moles of water, and *(iii)* the charge-balance condition of the aqueous solution.

Imposing the number of moles of the i-th species at equilibrium is obtained by defining the equilibrium constraint function as:

$$h(\mathbf{n}) := n_i - n_i^\star, \tag{2.25}$$

where $n_i^\star$ is the given number of moles for the i-th species. From this definition it follows that the gradient of h is given by:

$$\nabla h(\mathbf{n}) := \mathbf{e}_i, \tag{2.26}$$

with $\mathbf{e}_i \in \mathbb{R}^N$ denoting the vector whose i-th component is one, and all others zero.

Imposition of the charge-balance condition

As we just saw, the charge-balance condition is a convenient constraint to be imposed when the amount of a chemical element is unknown. By accounting for the balance of electrical charges in the system, the following condition must be attained:

$$\boldsymbol{z}^T \mathbf{n} = 0, \tag{2.27}$$

where $\boldsymbol{z} \in \mathbb{R}^N$ is the vector of charges of all species. Therefore, the equilibrium constraint function that imposes the charge-balance condition is given by:

$$h(\mathbf{n}) := \boldsymbol{z}^T \mathbf{n}, \tag{2.28}$$

whose gradient is:

$$\nabla h(\mathbf{n}) = \boldsymbol{z}. \tag{2.29}$$

Imposition of the activity of a species

Imposing the activity of a species can be useful in some cases. For example, the pH of an aqueous solution can be easily measured, and so the activity of species H^+ is known, given by:

$$a_{H^+} = \exp_{10}(-\text{pH}). \tag{2.30}$$

Therefore, this information can be used towards the calculation of the equilibrium state of the solution.

Imposing the activity of the i-th species requires defining the equilibrium constraint as:

$$h(\mathbf{n}) \coloneqq a_i(\mathbf{n}) - a_i^\star, \tag{2.31}$$

where $a_i^\star$ is the desired activity for the i-th species. It follows that the gradient of the above function is given by:

$$\nabla h(\mathbf{n}) = \nabla a_i(\mathbf{n}) \tag{2.32}$$

where $\nabla a_i(\mathbf{n})$ is the gradient of the activity function a_i. The analytical calculation of these derivatives results in efficient equilibrium calculations, and they are of utmost importance in the minimisation of the Gibbs free energy.

For simplicity, note that the dependence on temperature T and pressure P of the activity function a_i and its gradient ∇a_i in the previous formulation has been omitted.

Imposition of the partial pressure of a gaseous species

Imposing the partial pressure of a gas is a fairly common practice. In Bethke (2007), the equilibrium problem of the dissolution of pyrite (FeS_2) is considered. In this problem, the fugacity of the gaseous species O_2(g) is kept constant in order to simulate the contact of the solution with the atmosphere. Sometimes, however, one might opt to impose the partial pressure of a gas instead of its fugacity, which are equivalent practices when the gases are considered ideal. Thus, for instance, since oxygen makes up about 20% of the atmosphere, one needs to set the partial pressure of O_2(g) to 0.2 atm, assuming the atmospheric pressure is 1 atm.

The partial pressure of the i-th gaseous species is given by:

$$P_i = x_i^g P. \tag{2.33}$$

Thus, the equilibrium constraint function can be defined as:

$$h(\mathbf{n}) \coloneqq P_i - x_i^g P, \tag{2.34}$$

and its gradient as:

$$\nabla h(\mathbf{n}) \coloneqq -\nabla x_i^g P, \tag{2.35}$$

where the j-th component of the vector ∇x_i^g, denoted by $\partial x_i^g / \partial n_j$, should be zero if the j-th species does not belong to the gaseous phase. Note that x_i^g and ∇x_i^g can be easily calculated from $\mathbf{n}$ using the indices of the gaseous species $\mathcal{I}_g$.

2.3 Numerical Method

In this section the stoichiometric method to compute the equilibrium state of the system is presented, which requires the simultaneous solution of the non-linear mass-action equations (2.14) and the equilibrium constraint equations (2.20). By applying the natural logarithm to both sides of equations (2.14), the following equivalent system of equations can be written:

$$\mathbf{h}(\mathbf{n}) = \mathbf{0}, \tag{2.36}$$

$$\boldsymbol{\nu} \ln \mathbf{a}(\mathbf{n}) - \ln \mathbf{K} = \mathbf{0}. \tag{2.37}$$

Here, $\mathbf{K} \in \mathbb{R}^M$ is the vector of equilibrium constants of the reactions; and $\mathbf{a} \in \mathbb{R}^N$ is the vector of activities of the species.

2.3.1 Newton's Method

Equations (2.36) and (2.37) are solved with Newton's method, which is a well-known derivative-based root-finding algorithm that achieves second-order convergence near the solution (Nocedal and Wright, 1999). Therefore, a residual function $\mathbf{f}\colon \mathbb{R}^N \mapsto \mathbb{R}^N$ of the non-linear equations needs to be defined, which is here assumed as:

$$\mathbf{f}(\mathbf{n}) := \begin{bmatrix} \mathbf{h}(\mathbf{n}) \\ \boldsymbol{\nu} \ln \mathbf{a}(\mathbf{n}) - \ln \mathbf{K} \end{bmatrix}. \tag{2.38}$$

One can now define the problem as:

$$\begin{aligned} &\text{find } \mathbf{n} \text{ such that } \mathbf{f}(\mathbf{n}) = \mathbf{0}, \\ &\text{subject to } n_i > 0 \text{ for } i = 1, \ldots, N. \end{aligned} \tag{2.39}$$

The inequality constraint comes from the positivity condition of the molar amounts of the species. Note that $\mathbf{n}$ is also restricted to be component-wise non-zero, since every mass-action equation in (2.14) presumes the existence of every participating species in its corresponding equilibrium reaction in (2.13). As shown later, this restriction can be circumvented when a phase is not present in the equilibrium state.

The Jacobian function $\mathbf{J}\colon \mathbb{R}^N \mapsto \mathbb{R}^{N \times N}$, defined as $\mathbf{J} := \partial \mathbf{f} / \partial \mathbf{n}$, follows from equation (2.38) as:

$$\mathbf{J}(\mathbf{n}) = \begin{bmatrix} \nabla \mathbf{h}(\mathbf{n}) \\ \boldsymbol{\nu} \mathbf{A}(\mathbf{n}) \end{bmatrix}, \tag{2.40}$$

where the matrix of activity derivatives $\mathbf{A}$ is defined as:

$$\mathbf{A} := \frac{\partial \ln \mathbf{a}}{\partial \mathbf{n}}. \tag{2.41}$$

As seen in equation (2.41), the calculation of the Jacobian matrix $\mathbf{J}$ requires the first-order partial molar derivatives of the species activities $\mathbf{a}$.

The algorithm consists of iteratively calculating new approximations of $\mathbf{n}$ through the linear matrix equation:

$$\mathbf{J}_k \Delta \mathbf{n}_k = -\mathbf{f}_k, \tag{2.42}$$

where the subscript k denotes a quantity evaluated at the k-th iteration. Once $\Delta\mathbf{n}_k$ is determined, then $\mathbf{n}_{k+1}$ can be calculated as:

$$\mathbf{n}_{k+1} = \mathbf{n}_k + \Delta\mathbf{n}_k. \tag{2.43}$$

Note that the algorithm requires an initial guess $\mathbf{n}_0$, which should be near enough to the solution in order to prevent divergence.

2.3.2 Projection Procedure

In order to ensure that $\mathbf{n}_{k+1}$ in equation (2.43) is component-wise positive, it is projected to its feasible domain whenever one of its components become negative. This is done by slightly modifying equation (2.43) as:

$$n_i^{k+1} = n_i^k + \xi_i \Delta n_i^k \qquad (i = 1, \ldots, N), \tag{2.44}$$

where ξ_i is a projection factor given by:

$$\xi_i = \begin{cases} \dfrac{(\bar{\xi} - 1) n_i^k}{\Delta n_i^{k+1}} & \text{if } n_i^k + \Delta n_i^k \leq 0 \\ 1 & \text{otherwise} \end{cases} \qquad (i = 1, \ldots, N), \tag{2.45}$$

with $\bar{\xi} \in (0, 1)$ being a projection constant, and Δn_i^k is the Newton step along the i-th component at the k-th iteration.

The use of the projection factor in equation (2.45) guarantees that the positivity restriction on n_i^{k+1} is satisfied. This is done by replacing the i-th Newton step Δn_i^{k+1} by $(\bar{\xi} - 1) n_i^k$ whenever the former would lead the new iterate n_i^{k+1} to a negative state. Therefore, it is easy to see that the resulting equation becomes $n_i^{k+1} = \bar{\xi} n_i^k$, where not only the length of the Newton step has changed, but also its direction.

Numerical experiments indicate that the choice of $\bar{\xi}$ has a direct impact on the performance of the chemical equilibrium calculation. This is specially true when one phase in the system is absent in the final equilibrium state. Numerically, this is accompanied by the successive decrease of the total number of moles of the phase. A value of $\bar{\xi} = 0.1$ accelerates this disappearance process; however, if a phase is not supposed to vanish from the calculation, the use of a relatively small projection value can compromise the convergence speed of the method. Therefore, in order to balance efficiency and correct behaviour of the calculation, a projection constant $\bar{\xi}$ in the interval $[0.5, 0.7]$ is used. Morel and Morgan (1972) and Bethke (2007) use a similar approach, where $\bar{\xi}$ would correspond to a fixed value of 0.5 in the published algorithms.

2.3.3 Stabilisation Procedure

In Section 2.2, the chemical equilibrium problem for a multiphase system was formulated. However, depending on the conditions of temperature, pressure and equilibrium constraints, some phases might not be present at equilibrium. As a consequence, mass-action equations associated with non-existent phases will enforce unnecessary equilibrium conditions.

The literature contains some efficient approaches for handling phase disappearance in flash equilibrium calculations. For instance, the negative flash technique developed by Whitson and

Michelsen (1989) is a well-regarded classic. Nevertheless, it is difficult to adapt that technique to a stoichiometric methodology, since this approach does not use the Rachford–Rice equation commonly employed in flash calculations.

Determining the equilibrium phase assemblage of a multiphase system using the stoichiometric formulation is difficult and inelegant. However, unlike most geochemical solvers, this issue is not solved here with a trial and error approach of addition and removal of phases to determine the correct phase assemblage. Instead, a stabilisation procedure is applied to aid convergence of the stoichiometric method when potential phases in the calculation tend to disappearance. The developed heuristic technique consists of a simple modification in the activity derivative matrix $\mathbf{A}$ (see equation 2.41).

The difficulty in handling phase disappearance using the stoichiometric formulation results from the premise that the solution is an interior point of the feasible domain. In other words, the molar amounts of the species are positive. In Chapter 3, on the other hand, a non-stoichiometric formulation that presumes the possibility of a solution on the boundary of the feasible domain (i.e., where some phases and their species have zero number of moles) is presented. This is achieved by the introduction of extra unknowns into the minimisation problem, known as the *dual variables.* These appear when the Karush–Kuhn–Tucker (KKT) equations for non-linear programming problems are written.

The presentation of the stabilisation procedure requires the definition of an *unstable phase*.

Definition 2.4. Let n_t^{π} denote the total number of moles in phase π, and n_t the total number of moles in the system. The π-th phase in the system is said to be unstable, at iteration k, if the following conditions are satisfied:

$$n_t^{\pi,k} < \epsilon_s n_t^k \qquad \text{and} \qquad n_t^{\pi,k} < n_t^{\pi,k-1}, \tag{2.46}$$

where ϵ_s is a given stabilisation threshold value. In case $k = 0$, only the first condition is used. The threshold $\epsilon_s = 10^{-10}$ has proved to be satisfactory in the tests presented later.

The stabilisation procedure consists of assuming that:

$$\frac{\partial \ln a_i}{\partial n_j} = \frac{\delta_{ij}}{n_i} \qquad (i \in \mathcal{U},\ j = 1, \ldots, N), \tag{2.47}$$

where $\mathcal{U}$ denotes the set of indices of all chemical species that belongs to an unstable phase at the current iteration. As a consequence, the stabilised matrix $\bar{\mathbf{A}}$ in equation (2.41) is computed as follows:

$$\bar{A}_{ij} = \begin{cases} \dfrac{\delta_{ij}}{n_i} & \text{if } i \in \mathcal{U} \\ A_{ij} & \text{otherwise} \end{cases}, \tag{2.48}$$

where A_{ij} denotes the (i, j)-th entry in matrix $\mathbf{A}$; and δ_{ij} is the Kronecker delta.

This approach has the following effects: *(i)* for a multi-species phase, it eliminates the dependency of the activity derivative of a species with respect to the others; and *(ii)* for a single-species phase, it assumes a large activity derivative for the species, that would be zero otherwise, resulting in a decrease of the effect of the equilibrium condition imposed by the mass-action equations associated with that species.

Finally, note that the Jacobian matrix $\mathbf{J}$ may become ill-conditioned before a phase is detected to be unstable. In this work, a matrix $\mathbf{M}$ in a linear system $\mathbf{Mx} = \mathbf{b}$ is assumed to be ill-conditioned whenever the solution $\mathbf{x}$ does not satisfy:

$$\frac{\|\mathbf{Mx} - \mathbf{b}\|}{\|\mathbf{b}\| + 1} < \epsilon_{\text{ill}}, \tag{2.49}$$

for some small scalar ϵ_{ill}, where $\|\cdot\|$ denotes the ℓ^2-norm. In this case, the Newton step $\Delta\mathbf{n}_k$ is under-relaxed by a factor $\theta_{\text{ill}} \in (0, 1)$. Ill-conditioned cases have been successfully circumvented using $\theta_{\text{ill}} = 0.1$ and $\epsilon_{\text{ill}} = 10^{-8}$.

2.3.4 Convergence Criteria

The standard convergence criteria for Newton's method is adopted. Therefore, the iterative procedure should stop whenever the iterates satisfy:

$$\frac{\|\mathbf{n}_{k+1} - \mathbf{n}_k\|}{\|\mathbf{n}_{k+1}\| + 1} < \epsilon_{\mathbf{n}}, \tag{2.50}$$

and the residual vector fulfils the condition:

$$\|\mathbf{f}_{k+1}\| < \epsilon_{\mathbf{f}}, \tag{2.51}$$

where $\epsilon_{\mathbf{n}}$ and $\epsilon_{\mathbf{f}}$ are tolerance values.

However, the above criteria is not fulfilled when unstable phases exist. This happens because the unstable reactions (i.e., reactions associated with unstable phases) do not achieve equilibrium in the sense of satisfying their respective mass-action equations. As such, the residual contribution from these equations does not necessarily become small enough to pass the convergence test given by condition (2.51). Hence, a modification of the previous convergence criteria is required whenever unstable phases exist.

Let $\hat{\mathbf{f}}$ denote the residual vector $\mathbf{f}$ with entries set to zero whenever they correspond to a mass-action equation of an unstable reaction. Then, the proposed modified convergence criteria requires that condition (2.51) be satisfied with $\hat{\mathbf{f}}$ instead, so that unstable reactions are neglected in the convergence check.

CHAPTER 3

CHEMICAL EQUILIBRIUM: NON-STOICHIOMETRIC APPROACH

In this chapter a chemical equilibrium algorithm based on a non-stoichiometric approach is presented. The methodology consists of directly minimising the Gibbs free energy of a multiphase system, as opposed to the stoichiometric approach in Chapter 2, which solves a system of non-linear equations. Its main advantage is that it does not suffer from the difficulties encountered in the stoichiometric approach with respect to phase disappearances. This is because the minimisation approach adopted here requires no heuristic technique to determine the equilibrium phase assemblage.

3.1 Introduction

The geochemical literature presents several methodologies for minimising the Gibbs energy of multiphase systems. In this work, however, a novel approach is adopted, in which the Gibbs energy minimisation problem is transformed into a general non-linear non-convex programming problem. The objective of this abstraction is to simplify the problem formulation and propose its solution with the use of the rich set of optimisation tools developed by the mathematical community, which are in general rigorously proved to work and tested against hundreds of different problems.

This methodology has several advantages over the common approach of mixing optimisation and chemical concepts. Firstly, it is flexible enough to allow any optimisation algorithm to be tested for efficiency and robustness. Secondly, it eliminates the dependence of specific chemical details of the problem on its numerical solution, simplifying its analysis and implementation. Finally, it is easily extensible to allow any type of phases and species to be considered in the chemical system.

The numerical method for multiphase equilibrium calculations presented here is based on the trust-region primal-dual interior-point algorithm of Ulbrich et al. (2004) and Silva et al. (2008). Their algorithm solves non-linear programming problems with non-convex objective functions

containing both equality and inequality constraints. Their method is briefly described here for completeness. However, focus is given on our modifications, tailored to yield more efficient and robust chemical equilibrium calculations.

Although the proposed method can be used for standalone geochemistry modelling purposes, its development targets critical applications requiring sequential equilibrium calculations, such as reactive transport modelling in porous media flow. In this large-scale application, several local equilibrium calculations must be performed for each grid block of the mesh, at every time step. As a result, the chemical equilibrium method must efficiently perform each calculation in as few iterations as possible, using the minimum computational resources necessary.

Efficiency can be achieved, in part, by proper use of initial guesses. At each time step in a reactive flow, the molar amounts of the chemical elements are constantly varying at a fixed point due to the flux of chemical species. If the time step used in the temporal discretisation is sufficiently small, the variation of the local equilibrium state in a point should, in general, be small as well. Therefore, a good initial guess for the calculation of a local equilibrium state is its equilibrium state in a previous time level.

Using suitable initial guesses are, nevertheless, not sufficient to yield efficient calculations. An algorithm to be integrated into a reactive transport simulator requires at least superlinear rates of convergence near the solution. This will ensure convergence in only a few iterations when the initial guess is close to the solution, an important condition when millions to billions of equilibrium calculations have to be performed before moving to the next time level in the reactive flow calculations. As shown later, the interior-point algorithm of Ulbrich et al. (2004) attains, in practice, a linear rate of convergence near the solution, in spite of the use of exact Hessian expressions. Even though their algorithm is capable of finding a solution starting from very poor initial guesses, this low rate of convergence poses doubt on its acceptability for sequential equilibrium calculations.

In order to remedy these slow rates of convergence of Ulbrich et al. (2004) method, a *watchdog strategy* (Chamberlain et al., 1982) is adopted, which produces quadratic rates of convergence near the local optimum. The watchdog strategy consists of skipping all convergence strategies of Ulbrich et al. (2004) after some progress has been achieved in the calculation. However, if after some specified number of iterations the watchdog strategy was not able to improve the minimisation of the Gibbs free energy, than the algorithm of Ulbrich et al. (2004) is reactivated.

This hybrid approach has several benefits. As shown later, the number of iterations necessary for convergence in our examples decreased considerably. The watchdog strategy resulted in more efficient and robust calculations near a phase boundary, where phases can appear or disappear. Moreover, the use of computational resources is substantially decreased during the watchdog mode, since it skips several procedures that enforce a monotonic decrease in the optimality and feasibility measures of the calculation. Finally, recall that after the first calculation in a sequence of equilibrium calculations, a good initial guess is available for any subsequent computation. Therefore, in order to improve efficiency, the calculation starts with the use of the watchdog strategy, potentially yielding quadratic rates of convergence from the beginning.

The proposed method for minimising the Gibbs free energy in multiphase systems can only find local solutions. Therefore, solutions obtained by this method are not necessarily global

optima. However, recall that this algorithm is to be used for sequential equilibrium calculations in critical applications such as reactive transport simulations. Moreover, note that when the minimisation problem is non-convex, finding a global minimiser is a very expensive and time consuming calculation (Murty and Kabadi, 1987).

In order to resolve the problem of local versus global equilibrium points, initial guesses sufficiently close to the global minimum should be provided. Therefore, one would require a global minimum calculation only once, at the first time step, and expect that the subsequent steps only change the global minimum slightly. The proposed local minimisation algorithm, however, can be used in conjunction with global optimisation strategies to calculate a global optimum.

3.2 Governing Equations

Finding the equilibrium point of a multiphase system is a non-linear optimisation problem, where the Gibbs free energy of the system is minimised. The problem, however, is far from trivial, since it contains both *equality* and *inequality constraints*, defining, therefore, a non-linear programming problem.

The equality constraints arises from the need to specify, for example, the number of moles of each chemical element in the system, and the charge balance condition of an aqueous solution. The inequality constraints, on the other hand, results from the physical condition that the number of moles of the species are non-negative.

From the principle of minimum Gibbs free energy, the equilibrium state of a chemical system is calculated by solving the following constrained minimisation problem:

$$\min_{\mathbf{n}} G(\mathbf{n}; T, P) \quad \text{subject to} \begin{cases} \mathbf{h}(\mathbf{n}) = \mathbf{0} \\ \mathbf{n} \geq \mathbf{0} \end{cases}, \tag{3.1}$$

where $\mathbf{h}\colon \mathbb{R}^N \mapsto \mathbb{R}^C$ is the equilibrium constraint function; and C is the number of equilibrium constraints. The temperature T and pressure P are assumed given parameters.

3.2.1 Gibbs Energy Function

The Gibbs energy function $G\colon \mathbb{R}^{2+N} \mapsto \mathbb{R}$ is defined by:

$$G(T, P, \mathbf{n}) \coloneqq \sum_{i=1}^{N} n_i \mu_i(T, P, \mathbf{n}), \tag{3.2}$$

where $\mu_i\colon \mathbb{R}^{2+N} \mapsto \mathbb{R}$ is the chemical potential function of the i-th species, given by:

$$\mu_i(T, P, \mathbf{n}) = \mu_i^\circ(T, P) + RT \ln a_i(T, P, \mathbf{n}). \tag{3.3}$$

Similarly to the equilibrium constants of the reactions, the standard chemical potentials μ_i° of the species can also be interpolated from temperature versus pressure tables.

The chemical equilibrium method requires the exact gradient and Hessian of the Gibbs energy function. From equation (3.2), we write the first-order partial molar derivatives of the Gibbs energy function as:

$$\frac{\partial G}{\partial n_i} = \mu_i + \sum_{k=1}^{N} n_k \frac{\partial \mu_k}{\partial n_i} \qquad (i = 1, \ldots, N), \tag{3.4}$$

and its second-order partial molar derivatives as:

$$\frac{\partial^2 G}{\partial n_i \partial n_j} = \frac{\partial \mu_i}{\partial n_j} + \frac{\partial \mu_j}{\partial n_i} + \sum_{k=1}^{N} n_k \frac{\partial^2 \mu_k}{\partial n_i \partial n_j} \qquad (i, j = 1, \ldots, N). \tag{3.5}$$

For convenience reasons, the dependence on temperature T, pressure P, and composition $\mathbf{n}$ in the previous equations have been omitted.

As we shall see next, equations (3.4) and (3.5) will be simplified so that the Gibbs–Duhem equations are satisfied at equilibrium. This simplification has two main algorithmic advantages: *(i)* the Gibbs–Duhem equations are implicitly imposed, preventing, therefore, a large number of extra non-linear equality constraints in the optimisation problem, and *(ii)* it eliminates the need for the second-order partial molar derivatives of the chemical potentials in equation (3.5).

3.2.2 Gibbs–Duhem Equation

At equilibrium, any infinitesimal change in the system, under constant temperature and pressure, must satisfy the following differential equation:

$$\sum_{k=1}^{N} n_k \, \mathrm{d}\mu_k = 0, \tag{3.6}$$

which is known as the Gibbs–Duhem equation (Smith and Missen, 1982). This equation is, therefore, an additional constraint for the chemical equilibrium problem.

Depending on the equations of state used for the chemical potentials, there is no need to explicitly impose the Gibbs–Duhem constraint (3.6). This is because for these equations of state, condition (3.6) is satisfied at any composition point $\mathbf{n}$, even in disequilibrium. This happens, for example, when an ideal model is used for the computation of the activities of the species.

Sometimes, however, the use of some equations of state does not satisfy the Gibbs–Duhem equation (3.6) at every composition point $\mathbf{n}$. These equations of state are then said to be thermodynamically inconsistent. Nevertheless, in this particular situation, we need to make sure that the minimisation of the Gibbs free energy produces an equilibrium point where the Gibbs–Duhem constraint (3.6) is satisfied. However, equation (3.6) written in terms of total differentials is not adequate for this.

Based on the previous discussion, the following equivalent system of Gibbs–Duhem equations is used:

$$\sum_{k=1}^{N} n_k \frac{\partial \mu_k}{\partial n_i} = 0 \qquad (i = 1, \ldots, N). \tag{3.7}$$

Instead of imposing these conditions explicitly as equality constraints in the optimisation problem (3.1), we can enforce them implicitly in the gradient formula (3.4), which becomes:

$$\frac{\partial G}{\partial n_i} = \mu_i. \tag{3.8}$$

Explicitly imposing the non-linear Gibbs–Duhem equations results in $N + C$ equality constraints instead of C. This approach could be inefficient, since there would be an additional of N Lagrange multipliers in the problem, increasing by N (i.e., the number of species) the dimension of the linear systems.

Implicitly ensuring the Gibbs–Duhem equations (3.7), therefore, implies that:

$$\frac{\partial^2 G}{\partial n_i \partial n_j} = \frac{\partial \mu_i}{\partial n_j} \qquad (i, j = 1, \ldots, N). \tag{3.9}$$

Note, however, that this expression might not result in a symmetric Hessian matrix. This is because the choice of equations of state for the chemical potentials might not guarantee:

$$\frac{\partial \mu_i}{\partial n_j} = \frac{\partial \mu_j}{\partial n_i} \tag{3.10}$$

at any composition point $\mathbf{n}$. Nevertheless, this symmetry violation of the Hessian matrix causes no issues for the interior-point minimisation algorithm presented next. Following a similar analysis of Vanderbei and Shanno (1999), it can be shown that a sufficient condition for the Hessian matrix of the Gibbs energy function is that it is positive semi-definite.

3.3 Numerical Method

In this section a primal-dual interior-point algorithm is described for the chemical equilibrium calculation of multiphase systems. The description of the algorithm does not involve any specifics of the Gibbs energy minimisation problem. The idea is to separate pure mathematical concepts from those originating in the equilibrium problem.

Let us represent the Gibbs energy minimisation problem (3.1) as a general non-linear programming problem in the standard form:

$$\min_{\mathbf{x}} f(\mathbf{x}) \quad \text{subject to} \begin{cases} \mathbf{c}(\mathbf{x}) = \mathbf{0} \\ \mathbf{x} \geq \mathbf{0} \end{cases}, \tag{3.11}$$

where the objective function $f\colon \mathbb{R}^n \mapsto \mathbb{R}$ and the equality constraint function $\mathbf{c}\colon \mathbb{R}^n \mapsto \mathbb{R}^m$ are assumed to be twice continuously differentiable. Moreover, let us assume that $m \leq n$, where n denotes the number of variables and m the number of constraints. Finally, $\mathbf{x} \in \mathbb{R}^n$ denotes the vector of variables in the optimisation problem (3.11), which corresponds to the molar amounts of the chemical species in the context of the Gibbs free energy minimisation.

The solution of the non-linear programming problem (3.11) is summarised in Algorithm 3.1. Its detailed description can be found in the next sections.

3.3.1 First-Order Optimality Conditions

Let us write the necessary first-order optimality conditions for the non-linear programming problem (3.11). These are also known as the Karush–Kuhn–Tucker or KKT conditions (see Nocedal and Wright 1999), and depends on the following definition of the Lagrange function:

$$\mathcal{L}(\mathbf{x}, \mathbf{y}, \mathbf{z}) := f(\mathbf{x}) + \mathbf{c}(\mathbf{x})^T \mathbf{y} - \mathbf{x}^T \mathbf{z}, \tag{3.12}$$

Algorithm 3.1 Interior-Point Filter Trust-Region Algorithm

Require: $\mathbf{w}_0$, Δ_0°

$\mathbf{w}_k \leftarrow \mathbf{w}_0$
$\Delta_k^\circ \leftarrow \Delta_0^\circ$
for $k = 0$ to $k_{\max}$ **do**
  compute $\mathbf{s}_k^t$ and $\mathbf{s}_k^n$ using equations (3.39)–(3.42) at $\mathbf{w}_k$ ▷ *compute the tangential and normal steps*.
  compute M_k using equation (3.55) ▷ *compute the neighbourhood parameter* M_k.
  compute $\Delta_{k,j}^{\mathrm{n}}$ using equation (3.59) with $\mathbf{s}_k^t$ and $\mathbf{s}_k^n$, and Δ_k° ▷ *compute the largest neighbourhood trust-region radius*.
  if $\alpha^t(\Delta_k) < \epsilon_{\alpha_t}$ **then** ▷ *check if the safe tangential step needs to be calculated.*
    recompute $\mathbf{s}_k^t$ using equations (3.39) and (3.41) with $\sigma = \sigma^{\text{safe}}$.
    recompute Δ_k^{n} using equation (3.59) ▷ *this is necessary because* $\mathbf{s}_k^t$ has been updated.
  if $\theta(\mathbf{w}_k) > \Delta_k \min\{\gamma_1, \gamma_2 \Delta_k^\beta\}$ **then** ▷ *check if the restoration algorithm needs to be initiated.*
    start the *restoration algorithm* to produce $\mathbf{w}_{k+1}$ (see Algorithm 3.3).
  else
    start the *trust-region search algorithm* to produce $\mathbf{w}_{k+1}$ (see Algorithm 3.2).
  if $e(\mathbf{w}_{k+1}) \leq \epsilon_{\text{tol}}$ **then** ▷ *check for convergence, using equation* (3.66) *to compute* $e(\mathbf{w}_{k+1})$.
    stop ▷ *an optimum point has been found.*

where $\mathbf{y} \in \mathbb{R}^m$ and $\mathbf{z} \in \mathbb{R}^n$ are Lagrange multipliers. The KKT conditions, and their corresponding names, are then written as:

$$\nabla_x \mathcal{L}(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \mathbf{0}, \qquad \textit{optimality} \tag{3.13}$$

$$\mathbf{c}(\mathbf{x}) = \mathbf{0}, \qquad \textit{primal-feasibility} \tag{3.14}$$

$$\mathbf{X}\mathbf{z} = \mathbf{0}, \qquad \textit{complementarity} \tag{3.15}$$

$$\mathbf{x}, \mathbf{z} \geq \mathbf{0}, \qquad \textit{dual-feasibility} \tag{3.16}$$

where $\mathbf{X} \coloneqq \operatorname{diag}(\mathbf{x})$ and $\nabla_x \mathcal{L}$ denotes the gradient of the Lagrange function with respect to the primal variables $\mathbf{x}$, given by:

$$\nabla_x \mathcal{L}(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \nabla f(\mathbf{x}) + \nabla \mathbf{c}(\mathbf{x})^T \mathbf{y} - \mathbf{z}. \tag{3.17}$$

Equations (3.13)–(3.16) are the requirements that a local solution must satisfy. Note that besides the primal variables $\mathbf{x}$, the Lagrange multipliers $\mathbf{y}$ and $\mathbf{z}$ are also unknowns in the problem. Thus, there is a total of $2n + m$ unknowns, which corresponds to two times the number of chemical species plus the number of equilibrium constraints, $2N + C$. As discussed in Kulik et al. (2013), it is imperative that the size of the linear systems to be solved are moderate in order to efficiently solve equilibrium problems. Fortunately, the Lagrange multipliers $\mathbf{z}$ can be explicitly written in terms of $\mathbf{x}$ and $\mathbf{y}$, which decreases the size of the linear systems by n, as shown later.

3.3.2 Perturbed KKT Conditions

Similarly to El-Bakry et al. (1996), the primal-dual interior-point algorithm of Ulbrich et al. (2004) solves the problem (3.11) by suitably perturbing the KKT complementary condition (3.15). The approach results in the following perturbed system of non-linear equations:

$$\mathbf{F}_{\hat{\mu}}(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \mathbf{0}, \tag{3.18}$$

where $\mathbf{F}_{\hat{\mu}}$ is defined by:

$$\mathbf{F}_{\hat{\mu}}(\mathbf{x}, \mathbf{y}, \mathbf{z}) \coloneqq \begin{bmatrix} \nabla_x \mathcal{L}(\mathbf{x}, \mathbf{y}, \mathbf{z}) \\ \mathbf{c}(\mathbf{x}) \\ \mathbf{X}\mathbf{z} - \hat{\mu}\mathbf{e} \end{bmatrix}, \tag{3.19}$$

with $\hat{\mu} > 0$ denoting a small perturbation parameter; and $\mathbf{e} \in \mathbb{R}^n$ the vector of all ones.

Equation (3.18) is solved using Newton's method. However, several strategies are adopted to aid convergence from arbitrary or poor initial guesses, since Newton's method is not guaranteed to converge far from a local solution (Nocedal and Wright, 1999). Other strategies are also adopted to guarantee that the solution is a local minimum, and not a maximum for example. These will be presented later.

At every iteration, the perturbation parameter $\hat{\mu}$ is appropriately decreased, so that the solution of the original problem is obtained when $\hat{\mu} \to 0$. Therefore, applying Newton's method to equation (3.18) results in:

$$\mathbf{J}(\mathbf{w})\Delta\mathbf{w} = -\mathbf{F}_{\hat{\mu}}(\mathbf{w}), \tag{3.20}$$

where $\mathbf{w} \coloneqq (\mathbf{x}, \mathbf{y}, \mathbf{z})$, and $\mathbf{J}(\mathbf{w})$ is the Jacobian of $\mathbf{F}_{\hat{\mu}}(\mathbf{w})$, given by:

$$\mathbf{J}(\mathbf{w}) \coloneqq \begin{bmatrix} \nabla^2_{xx}\mathcal{L}(\mathbf{w}) & \nabla\mathbf{c}(\mathbf{x})^T & -\mathbf{I} \\ \nabla\mathbf{c}(\mathbf{x}) & \mathbf{0} & \mathbf{0} \\ \mathbf{Z} & \mathbf{0} & \mathbf{X} \end{bmatrix}, \tag{3.21}$$

with $\mathbf{I}$ denoting the $n \times n$ identity matrix; $\mathbf{Z} \coloneqq \operatorname{diag}(\mathbf{z})$; and $\nabla^2_{xx}\mathcal{L}(\mathbf{w})$ the Hessian of the Lagrange function:

$$\nabla^2_{xx}\mathcal{L}(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \nabla^2 f(\mathbf{x}) + \sum_{j=1}^{M} \nabla^2 c_j(\mathbf{x})^T y_j. \tag{3.22}$$

Note that the Jacobian matrix $\mathbf{J}(\mathbf{w})$ does not depend on the perturbation parameter $\hat{\mu}$.

3.3.3 Convergence Strategies

This section presents the convergence strategies implemented in the algorithm to aid convergence from poor initial guesses. A more mathematical in-depth description of these strategies can be found in Ulbrich et al. (2004) and Silva et al. (2008).

Filter

The filter technique developed by Fletcher and Leyffer (2002) is adopted in the algorithm. The idea has its origins in multi-criteria optimisation problems, whose adaptation for non-linear programming problems is possible by considering the minimisation of the *optimality* and *feasibility measures* as two competing targets. These measures are related to the norm of the optimality and feasibility conditions (3.13) and (3.14) respectively, and are defined next.

The filter works as follows. At every iteration, a decision is made about storing or not the current optimality and feasibility measures. These records are then used in subsequent iterations to decide the rejection of poor iterates. A poor iterate is defined as an iterate whose corresponding optimality and feasibility measures are not sufficiently smaller than those stored in the filter. Therefore, the filter guarantees that at every iteration the current iterate is closer to a local optimum solution.

In order to describe the filter technique, let us introduce the following measures:

$$\theta_h(\mathbf{w}) := \|\mathbf{c}(\mathbf{x})\|, \tag{3.23}$$

$$\theta_c(\mathbf{w}) := \|\mathbf{X}\mathbf{z} - \mu\mathbf{e}\|, \tag{3.24}$$

$$\theta_l(\mathbf{w}) := \|\nabla_x \mathcal{L}(\mathbf{w})\|. \tag{3.25}$$

Next, define, respectively, the feasibility and optimality measures that comprise the entries of the filter:

$$\theta(\mathbf{w}) := \theta_c(\mathbf{w}) + \theta_h(\mathbf{w}) \tag{3.26}$$

$$\psi(\mathbf{w}) := f(\mathbf{x}) + c_\psi \mu, \tag{3.27}$$

where $c_\psi > 0$ is a constant calculated during the initialisation of the algorithm and defined as:

$$c_\psi := \frac{3n^2}{1-\sigma_{\min}} \left[\max\left\{1, \frac{1-\sigma_{\min}}{\gamma}\right\}\right]^2, \tag{3.28}$$

where γ is given later by equation (3.53).

Note that in the original work of Ulbrich et al. (2004), the optimality measure $\psi(\mathbf{w})$ was defined by:

$$\psi(\mathbf{w}) := \mu + \|\nabla_x \mathcal{L}(\mathbf{w})\|^2. \tag{3.29}$$

However, Silva et al. (2008) slightly modified the algorithm to allow the use of measure (3.27) instead, which is based on the objective function, rather than the gradient of the Lagrange function. Silva et al. (2008) also presents the following alternative measure based on the Lagrange function:

$$\psi(\mathbf{w}) := \mathcal{L}(\mathbf{w}) + (c_\psi + n)\mu. \tag{3.30}$$

It can be shown that for small values of μ and near the solution, the latter definition tends to (3.27). Therefore, in this work the definition (3.27) is adopted for $\psi(\mathbf{w})$.

Let us now describe the filter concept. In this algorithm the filter $\mathcal{F}$ is defined as a finite set of pairs:

$$\mathcal{F} := \{(\theta(\mathbf{w}_i), \psi(\mathbf{w}_i))\}, \tag{3.31}$$

where i denotes the iteration number at which the measures $\theta(\mathbf{w}_i)$ and $\psi(\mathbf{w}_i)$ were evaluated.

The filter $\mathcal{F}$ is required not to contain any dominated entry. In other words, for any entry $(\theta(\mathbf{w}_j), \psi(\mathbf{w}_j)) \in \mathcal{F}$, there is no other entry $(\theta(\mathbf{w}_i), \psi(\mathbf{w}_i)) \in \mathcal{F}$ that satisfies:

$$\theta(\mathbf{w}_j) \leq \theta(\mathbf{w}_i) \qquad \text{and} \qquad \psi(\mathbf{w}_j) \leq \psi(\mathbf{w}_i). \tag{3.32}$$

If condition (3.32) is true, then entry $(\theta(\mathbf{w}_i), \psi(\mathbf{w}_i))$ is said to be dominated by $(\theta(\mathbf{w}_j), \psi(\mathbf{w}_j))$. This requirement is enforced every time a new entry is added to the filter by removing any existing entry dominated by the new one.

At every iteration, the filter scheme discards any trial iterate $\mathbf{w}'$ that is not acceptable by $\mathcal{F}$. The point $\mathbf{w}'$ is said to be acceptable by the filter $\mathcal{F}$ if for every entry $(\hat{\theta}, \hat{\psi}) \in \mathcal{F}$ the following hold:

$$\theta(\mathbf{w}') < (1 - \alpha_\theta)\hat{\theta} \qquad \text{or} \qquad \psi(\mathbf{w}') < \hat{\psi} - \alpha_\psi \hat{\theta}, \tag{3.33}$$

where $\alpha_\theta, \alpha_\psi \in (0, 1)$ are suitable small constants used to prevent acceptance of points located considerably close to the boundary of the filter (see Table 3.1). In this way, $\mathbf{w}'$ has to provide a sufficient decrease in the measures θ and ψ to be acceptable.

Trust-Region Approach

Newton's method frequently behaves erratically far from the solution. Thus, convergence strategies exist in which the length of the step taken towards a new iterate is somehow controlled. Differently from a line-search strategy, in which the size of Newton's step is controlled and its direction preserved, a trust-region strategy controls both its size and direction. This is done with the introduction of a decomposition of Newton's step into a *tangential* and *normal* components, whose lengths are restricted so that they fit inside a variable trust-region.

The benefit of this decomposition is the possibility to control how much the iterations should favour the decrease of either the feasibility measure θ or the optimality measure ψ (see equations 3.26 and 3.27). By controlling the size of the tangential step, iterates that satisfy more the feasibility condition (3.14) are obtained. For example, the smaller the feasibility measure, the smaller the mass-balance residuals in the minimisation of Gibbs free energy. On the other hand, by controlling the size of the normal step, iterates that conform more to the optimal condition (3.13) are obtained. In addition, the lengths of these components can be appropriately adjusted to enforce positive number of moles at every iteration.

Similarly to Ulbrich et al. (2004), the tangential and normal steps $\mathbf{s}^t$ and $\mathbf{s}^n$ are defined by the following equations:

$$\mathbf{J}(\mathbf{w})\mathbf{s}^t = -\begin{bmatrix} \nabla_x \mathcal{L}(\mathbf{w}) \\ \mathbf{0} \\ (1 - \sigma)\mu\mathbf{e} \end{bmatrix} \tag{3.34}$$

and

$$\mathbf{J}(\mathbf{w})\mathbf{s}^n = -\begin{bmatrix} \mathbf{0} \\ \mathbf{c}(\mathbf{x}) \\ \mathbf{X}\mathbf{z} - \mu\mathbf{e} \end{bmatrix}, \tag{3.35}$$

where the perturbed parameter $\hat{\mu}$ of equation (3.19) has been defined as $\hat{\mu} = \sigma\mu$, with $\sigma \in (0, 1)$ being a centering parameter and μ being a measure of complementarity given by:

$$\mu \coloneqq \frac{\mathbf{x}^T\mathbf{z}}{n}. \tag{3.36}$$

From Silva et al. (2008), the centering parameter σ is calculated using:

$$\sigma = \begin{cases} \sigma_{\max} & \text{if } \mu < \mu_\sigma \\ \sigma_{\min} & \text{otherwise} \end{cases}, \tag{3.37}$$

where $\sigma_{\min}$, $\sigma_{\max}$ and μ_σ are positive constants. Their values adopted in this work are given in Table 3.1. Note that, by construction, the tangential and normal steps satisfies $\Delta\mathbf{w} = \mathbf{s}^t + \mathbf{s}^n$.

It can be shown that the primal-dual step $\Delta\mathbf{w}$ is a descent direction for $\mathbf{x}^T\mathbf{z}/n$, implying a dynamic reduction of μ as the calculations proceed (Ulbrich et al., 2004). This choice is commonly adopted in linear programming algorithms, though it was also used by El-Bakry et al. (1996) in his non-linear programming method.

Nocedal et al. (2009) presents an adaptive strategy for the update of the σ parameter based on the minimisation of an one-dimensional quality function. In the optimisation code LOQO (Vanderbei and Shanno, 1999; Vanderbei, 1999), a heuristic equation for the dynamic update of the centering parameter σ is also presented. Both approaches were tested, but in general they required several more iterations to converge than the formula (3.37) of Silva et al. (2008), though with different parameter values tested with chemical equilibrium problems.

In practice, the linear systems (3.34) and (3.35) need to be simplified for efficiency reasons. Let $\bar{\mathbf{J}}(\mathbf{w})$ denote the following reduced Jacobian matrix:

$$\bar{\mathbf{J}}(\mathbf{w}) := \begin{bmatrix} \nabla^2_{xx}\mathcal{L}(\mathbf{w}) + \mathbf{X}^{-1}\mathbf{Z} & \nabla\mathbf{c}(\mathbf{x})^T \\ \nabla\mathbf{c}(\mathbf{x}) & \mathbf{0} \end{bmatrix}. \tag{3.38}$$

Let $\mathbf{s}^t = (\mathbf{s}^t_x, \mathbf{s}^t_y, \mathbf{s}^t_z)^T$ and $\mathbf{s}^n = (\mathbf{s}^n_x, \mathbf{s}^n_y, \mathbf{s}^n_z)^T$. It can be shown that solving (3.34) and (3.35) is equivalent to solving:

$$\bar{\mathbf{J}}(\mathbf{w}) \begin{bmatrix} \mathbf{s}^t_x \\ \mathbf{s}^t_y \end{bmatrix} = - \begin{bmatrix} \nabla_x\mathcal{L}(\mathbf{w}) + (1-\sigma)\mu\mathbf{X}^{-1}\mathbf{e} \\ \mathbf{0} \end{bmatrix} \tag{3.39}$$

and

$$\bar{\mathbf{J}}(\mathbf{w}) \begin{bmatrix} \mathbf{s}^n_x \\ \mathbf{s}^n_y \end{bmatrix} = - \begin{bmatrix} \mathbf{z} - \mu\mathbf{X}^{-1}\mathbf{e} \\ \mathbf{c}(\mathbf{x}) \end{bmatrix}, \tag{3.40}$$

and then, using the previous results to calculate:

$$\mathbf{s}^t_z = -\mathbf{X}^{-1}[\mathbf{Z}\mathbf{s}^t_x + (1-\sigma)\mu\mathbf{e}] \tag{3.41}$$

and

$$\mathbf{s}^n_z = -\mathbf{X}^{-1}[\mathbf{Z}\mathbf{s}^n_x + \mathbf{X}\mathbf{z} - \mu\mathbf{e}]. \tag{3.42}$$

This simplification results in the solution of linear systems with dimension $n + m$, instead of $2n + m$. Observe that the coefficient matrix $\bar{\mathbf{J}}(\mathbf{w})$ is the same for both linear systems. This allows the calculation of a decomposition of $\bar{\mathbf{J}}(\mathbf{w})$ once, and then its repeated use for efficient solution of the linear systems (3.39) and (3.40).

The algorithm of Ulbrich et al. (2004) introduces a trust-region radius Δ in order to control the length of the tangential and normal steps $\mathbf{s}^t$ and $\mathbf{s}^n$. At every iteration, the damped tangential and normal steps $\alpha^t(\Delta)\mathbf{s}^t$ and $\alpha^n(\Delta)\mathbf{s}^n$ satisfy:

$$\|\alpha^n(\Delta)\mathbf{s}^n\| \leq \Delta \qquad \text{and} \qquad \|\alpha^t(\Delta)\mathbf{s}^t\| \leq \Delta, \tag{3.43}$$

where the damping factors $\alpha^n(\Delta)$ and $\alpha^t(\Delta)$ are given by:

$$\alpha^n(\Delta) := \min\left\{1, \frac{\Delta}{\|\mathbf{s}^n\|}\right\} \tag{3.44}$$

Table 3.1: Parameters of the interior-point minimisation algorithm.

Parameter	Value	Parameter	Value
Filter		**Safe Tangential Step**	
α_θ	10^{-3}	$\epsilon_1^{\alpha_t}$	0.8
α_ψ	10^{-3}	$\epsilon_2^{\alpha_t}$	10^{-3}
Central Neighbourhood		$\sigma_{\min}^{\text{safe}}$	0.1
M°	10^3	$\sigma_{\max}^{\text{safe}}$	0.5
α_M°	10^3	**Restoration**	
α_M	10	ξ_1	10^{-5}
ϵ_M	10^{-3}	ξ_2	0.5
γ°	10^{-3}	γ_1	0.5
Trust-Region		γ_2	1
Δ_0°	10^5	β	0.75
$\Delta_{\min}$	10^{-12}	σ^{rest}	1
κ	0.1	**Watchdog**	
η_1	10^{-4}	$\dot{\mu}$	10^{-15}
η_2	0.8	μ_w	10^{-2}
Initial Guess		W	10
$x_0^{\min}$	10^{-14}	$\psi_{\max}$	10^{-1}
$z_0^{\min}$	10^{-10}	τ	0.995
Sigma			
μ_σ	10^{-6}		
$\sigma_{\min}$	0.1		
$\sigma_{\max}$	0.5		

and

$$\alpha^t(\Delta) \coloneqq \min\left\{\alpha^n(\Delta), \frac{\Delta}{\|\mathbf{s}^t\|}\right\}. \tag{3.45}$$

The above definition imposes $\alpha^t(\Delta) \leq \alpha^n(\Delta)$, which is used to enforce the iterates to stay in the central neighbourhood $\mathcal{N}(\gamma, M_k)$ defined in (3.52).

Note that Ulbrich et al. (2004) and Silva et al. (2008) presume a well-scaled problem with the definition of a circular trust-region given by equation (3.43). Thus it is advised to apply proper scaling of the variables when using their algorithm.

With the introduction of the trust-region radius Δ, and the damping factors $\alpha^n(\Delta)$ and $\alpha^t(\Delta)$, a trial iterate at the k-th iteration $\mathbf{w}_k(\Delta)$ is defined as:

$$\mathbf{w}_k(\Delta) \coloneqq \mathbf{w}_k + \alpha^t(\Delta)\mathbf{s}_k^t + \alpha^n(\Delta)\mathbf{s}_k^n. \tag{3.46}$$

Ulbrich et al. (2004) and Silva et al. (2008) then defines the following trust-region model function to approximate the optimality measure $\psi(\mathbf{w})$ at the trial iterate:

$$m_k(\mathbf{w}_k(\Delta)) \coloneqq \psi(\mathbf{w}_k) + \nabla\psi(\mathbf{w}_k)^T(\mathbf{w}_k(\Delta) - \mathbf{w}_k). \tag{3.47}$$

This model function is used as a reference to impose a sufficient decrease in the optimality measure $\psi(\mathbf{w})$. For this, the actual and predicted decrease of the optimality measure are defined by:

$$\Delta\psi_{k,\text{actual}} \coloneqq \psi(\mathbf{w}_k) - \psi(\mathbf{w}_k(\Delta)) \tag{3.48}$$

Algorithm 3.2 Trust-Region Search Algorithm

Require: $\mathbf{w}_k$, Δ_k^{n}, $\mathbf{s}_k^t$ and $\mathbf{s}_k^n$

$\mathbf{w}_k \leftarrow \mathbf{w}_0$
$\Delta_k \leftarrow \Delta_k^{\mathrm{n}}$
loop
  if $\mathbf{w}_k(\Delta_k)$ is acceptable by $\mathcal{F}$ **then** ▷ *check if trial iterate $\mathbf{w}_k(\Delta_k)$ is acceptable by the filter.*
    $\rho(\Delta_k) \leftarrow \dfrac{\psi(\mathbf{w}_k) - \psi(\mathbf{w}_k(\Delta_k))}{m_k(\mathbf{w}_k) - m_k(\mathbf{w}_k(\Delta_k))}$ ▷ *compute the ratio of actual to predicted decrease of ψ.*
    if $m_k(\mathbf{w}_k) - m_k(\mathbf{w}_k(\Delta_k)) \leq \kappa\theta(\mathbf{w}_k)^2$ **then**
      $\mathcal{F} \leftarrow \mathcal{F} \cup (\theta(\mathbf{w}_k), \psi(\mathbf{w}_k))$ ▷ *extend the filter and remove from it any dominated entry.*
      $\Delta_{k+1}^{\circ} \leftarrow 2\Delta_k$ ▷ *expand the trust-region.*
      return $\mathbf{w}_k(\Delta_k)$ ▷ *accept the trial iterate $\mathbf{w}_k(\Delta_k)$ as $\mathbf{w}_{k+1}$.*
    if $\rho(\Delta_k) > \eta_1$ **then** ▷ *check if a sufficient decrease in ψ has been achieved.*
      $\Delta_{k+1}^{\circ} \leftarrow \begin{cases} 2\Delta_k & \text{if } \rho(\Delta_k) > \eta_2 \\ \Delta_k & \text{otherwise} \end{cases}$ ▷ *expand the trust-region if possible.*
      return $\mathbf{w}_k(\Delta_k)$ ▷ *accept the trial iterate $\mathbf{w}_k(\Delta_k)$ as $\mathbf{w}_{k+1}$.*
  $\Delta_k \leftarrow \Delta_k/2$ ▷ *contract the trust-region.*
  if $\Delta_k \leq \Delta_{\min}$ **then**
    error *"could not find a trust-region radius that sufficiently decrease ψ and produces an iterate acceptable by $\mathcal{F}$."*

and

$$\Delta\psi_{k,\mathrm{pred}} \coloneqq m_k(\mathbf{w}_k) - m_k(\mathbf{w}_k(\Delta)). \tag{3.49}$$

Let $\rho(\Delta) \coloneqq \Delta\psi_{k,\mathrm{actual}}/\Delta\psi_{k,\mathrm{pred}}$. At every iteration, we enforce:

$$\rho(\Delta) \geq \eta, \tag{3.50}$$

for a predefined $\eta \in (0,1)$, so that the current trust-region with radius Δ defines a region where the model m is a good approximation to the optimality measure $\psi(\mathbf{w})$. Accordingly to the value of $\rho(\Delta)$, the algorithm either contracts or expands the trust-region.

The previous test is skipped, however, if the following condition holds:

$$\Delta\psi_{k,\mathrm{pred}} < \kappa\theta(\mathbf{w}_k)^2, \tag{3.51}$$

where $\kappa \in (0,1)$ is a given constant (see Table 3.1). In this case, the trial iterate $\mathbf{w}_k(\Delta)$ is accepted and we set $\mathbf{w}_{k+1} = \mathbf{w}_k(\Delta)$. Moreover, the filter $\mathcal{F}$ is extended with $(\theta(\mathbf{w}_k), \psi(\mathbf{w}_k))$. Recall that any existing entry in $\mathcal{F}$ that is dominated by the new entry must be excluded.

Algorithm 3.2 presents a search algorithm for finding a suitable trust-region radius Δ that satisfies the previous conditions.

Central Neighbourhood

The primal-dual interior-point algorithm of Ulbrich et al. (2004) uses a technique to prevent the iterates $\mathbf{x}_k(\Delta)$ and $\mathbf{z}_k(\Delta)$ from approaching their bounds too rapidly. The idea consists of

controlling the trust-region radius Δ at every iteration so that $\mathbf{w}_k(\Delta) \in \mathcal{N}(\gamma, M_k)$, where:

$$\mathcal{N}(\gamma, M_k) \coloneqq \{\mathbf{w} : \mathbf{x}, \mathbf{z} > \mathbf{0}, \quad \mathbf{X}\mathbf{z} \geq \gamma\mu\mathbf{e}, \quad \theta(\mathbf{w}) \leq M_k\mu\}, \tag{3.52}$$

with $\mu = \mathbf{x}^T\mathbf{z}/n$. The neighbourhood constant γ is calculated during the initialisation stage of the algorithm, given by:

$$\gamma \coloneqq \min\left\{\gamma^\circ, \frac{1}{2\mu_0}\min_i\{x_i^0 z_i^0\}\right\}, \tag{3.53}$$

where γ° is a positive constant; x_i^0 and z_i^0 are the i-th components of the initial guesses $\mathbf{x}_0$ and $\mathbf{z}_0$; and $\mu_0 = \mathbf{x}_0^T\mathbf{z}_0/n$ is the initial perturbation parameter. The neighbourhood parameter M_0 is given by:

$$M_0 = \max(M^\circ, \alpha_M^\circ\theta(\mathbf{w}_0)/\mu_0), \tag{3.54}$$

and for $k > 0$ we have:

$$M_{k+1} = \begin{cases} \max(M^\circ, \alpha_M\theta(\mathbf{w}_k)/\mu_k) & \text{if } \theta(\mathbf{w}_k) > \mu_k\epsilon_M M_k \\ M_k & \text{otherwise} \end{cases}, \tag{3.55}$$

where M°, α_M, α_M°, and ϵ_M are positive constants. See Table 3.1 for their values.

Section C.1 describes the procedure to calculate the *largest feasible trust-region radius* Δ_k^{f} that satisfies the first central neighbourhood condition:

$$\mathbf{x}_k(\Delta), \mathbf{z}_k(\Delta) > \mathbf{0}. \tag{3.56}$$

Section C.2 shows the approach for calculating the *largest central trust-region radius* Δ_k^{c} that satisfies the second central neighbourhood condition:

$$\mathbf{X}_k(\Delta)\mathbf{z}_k(\Delta) \geq \gamma\mu_k(\Delta)\mathbf{e}. \tag{3.57}$$

Note, however, that calculating the *largest neighbourhood trust-region radius* Δ_k^{n} that satisfies the third central neighbourhood condition:

$$\theta(\mathbf{w}_k(\Delta)) \leq M_k\mu_k(\Delta), \tag{3.58}$$

cannot be done explicitly. Its value is approximated by:

$$\Delta_k^{\mathrm{n}} = \frac{1}{2^i}\min\{\Delta_k^{\mathrm{f}}, \Delta_k^{\mathrm{c}}, \Delta_k^\circ\}, \tag{3.59}$$

where $i \in \mathbb{N}$ is the smallest number that results in Δ_k^{n} satisfying (3.58). The parameter Δ_k° is the finite upper-bound value that the trust-region radius can assume at the k-th iteration. By construction, it can be stated that Δ_k^{n} satisfies all the three central neighbourhood conditions in equation (3.52).

Restoration Phase

Whenever the algorithm fails to find an iterate that does not satisfy the filter and trust-region conditions, some action other than halt its execution must be performed. When this happens, the algorithm enters in a restoration phase, in which it focuses on the decrease of an alternative feasibility measure. During the iterations in the restoration phase, some progress towards

improved optimality measures is also obtained, though the main result after the end of the restoration phase is in general the decrease of the feasibility measure (e.g., the mass-balance residuals). Once an iterate in the restoration phase satisfies the filter conditions, the normal trust-region algorithm takes back control of the calculation.

From Ulbrich et al. (2004), the restoration phase algorithm is initiated whenever:

$$\theta(\mathbf{w}_k) > \Delta_k \min\{\gamma_1, \gamma_2 \Delta_k^{\beta}\}, \tag{3.60}$$

where γ_1, γ_2 and β are predefined positive constants (see Table 3.1). The filter is extended before the start of the restoration algorithm with the pair $(\theta(\mathbf{w}_k), \psi(\mathbf{w}_k))$. This prevents possible subsequent iterates $\mathbf{w}'$ with corresponding $\theta(\mathbf{w}')$ very close to $\theta(\mathbf{w}_k)$, which could potentially initiate the restoration phase again.

The idea of the restoration phase algorithm is to find an iterate $\mathbf{w}_{k+1}$ that *(i)* is acceptable by the filter $\mathcal{F}$, and *(ii)* satisfies the condition:

$$\theta(\mathbf{w}_{k+1}) \leq \Delta_k \min\{\gamma_1, \gamma_2 \Delta_k^{\beta}\}. \tag{3.61}$$

After $\mathbf{w}_{k+1}$ has been found, the calculation returns to the main algorithm. However, the infeasibility of the restored iterate $\mathbf{w}_{k+1}$ is considerably smaller.

Let us define the *restoration measure* $\theta_2(\mathbf{w})$ as:

$$\theta_2(\mathbf{w}) := \frac{1}{2}(\theta_h(\mathbf{w})^2 + \theta_c(\mathbf{w})^2). \tag{3.62}$$

It can be shown that the tangential and normal steps $\mathbf{s}^t$ and $\mathbf{s}^n$ produced by equations (3.34) and (3.35) satisfy:

$$\nabla\theta_2(\mathbf{w})^T \mathbf{s}^t = 0 \quad \text{and} \quad \nabla\theta_2(\mathbf{w})^T \mathbf{s}^n = -2\theta_2(\mathbf{w}). \tag{3.63}$$

From these results, it can be stated that the normal step $\mathbf{s}^n$ is a *descent direction* for $\theta_2(\mathbf{w})$, since $\nabla\theta_2(\mathbf{w})^T \mathbf{s}^n < 0$.

The restoration algorithm, therefore, consists of computing the normal and tangential steps as before. However, its purpose is to minimise the restoration measure $\theta_2(\mathbf{w})$, rather than the optimality measure $\psi(\mathbf{w})$. This has implications on the search for suitable trust-region radius at every iteration of the restoration algorithm. For instance, the accepted trust-region radius Δ must yield a sufficient decrease on the measure $\theta_2(\mathbf{w})$.

Algorithm 3.3 presents the restoration algorithm based on the previous discussion. If successful, it produces an iterate $\mathbf{w}_{k+1}$ that is acceptable by the filter $\mathcal{F}$ and satisfies condition (3.61).

Safe Tangential Step

Silva et al. (2008) introduced a safe tangential step procedure in their interior-point algorithm. The approach recomputes $\mathbf{s}^t$ using equations (3.39) and (3.41) with $\sigma = \sigma^{\text{safe}}$ whenever:

$$\alpha^t(\Delta_k) < \epsilon_1^{\alpha_t}, \tag{3.64}$$

for $\epsilon_1^{\alpha_t} \in (0, 1)$. The calculation of σ^{safe} is given by:

$$\sigma^{\text{safe}} = \begin{cases} \sigma_{\max}^{\text{safe}} & \text{if } \alpha(\Delta) < \epsilon_2^{\alpha_t} \\ \sigma_{\min}^{\text{safe}} & \text{otherwise} \end{cases}, \tag{3.65}$$

Algorithm 3.3 Restoration Algorithm

Require: $\mathbf{w}_k$

$\mathcal{F} \leftarrow \mathcal{F} \cup (\theta(\mathbf{w}_k), \psi(\mathbf{w}_k))$ ▷ *extend the filter and remove from it any dominated entry.*

$j \leftarrow 0$ ▷ *the iteration counter in the restoration loop.*

$\mathbf{w}_{k,j} \leftarrow \mathbf{w}_k$ ▷ *the iterate at the j-th iteration.*

$\Delta^{\circ}_{k,j} \leftarrow \Delta^{\circ}_k$ ▷ *the upper-bound on the trust-region radius at the j-th iteration.*

while $\mathbf{w}_{k,j}$ is not acceptable to $\mathcal{F}$ **and** $\theta(\mathbf{w}_{k,j}) > \Delta_k \min\{\gamma_1, \gamma_2 \Delta_k^{\beta}\}$ **do**

  compute $\mathbf{s}^t_{k,j}$ and $\mathbf{s}^n_{k,j}$ using equations (3.39)–(3.42) at $\mathbf{w}_{k,j}$ using $\sigma = \sigma^{\text{rest}}$

  compute $\Delta^{\text{n}}_{k,j}$ using equation (3.59) with $\mathbf{s}^t_{k,j}$ and $\mathbf{s}^n_{k,j}$, and $\Delta^{\circ}_{k,j}$

  $\Delta_{k,j} \leftarrow \Delta^{\text{n}}_{k,j}$ ▷ *set the trial trust-region radius at the j-th iteration.*

  repeat

    $\rho(\Delta_{k,j}) \leftarrow \dfrac{\theta_2(\mathbf{w}_{k,j}) - \theta_2(\mathbf{w}_{k,j}(\Delta_{k,j}))}{2\theta_2(\mathbf{w}_{k,j})}$ ▷ *compute the ratio of actual to predicted decrease of* θ_2.

    $\Delta_{k,j} \leftarrow \begin{cases} \Delta_{k,j}/2 & \text{if } \rho(\Delta_{k,j}) < \xi_1 \\ \Delta_{k,j} & \text{otherwise} \end{cases}$ ▷ *contract the trust-region if another inner iteration is necessary.*

  until $\rho(\Delta_{k,j}) \geq \xi_1$ ▷ *check if a sufficient decrease in* θ_2 *has been achieved.*

  $\Delta^{\circ}_{k,j+1} \leftarrow \begin{cases} 2\Delta^{\circ}_{k,j} & \text{if } \rho(\Delta_{k,j}) > \xi_2 \\ \Delta^{\circ}_{k,j} & \text{otherwise} \end{cases}$ ▷ *allow the trust-region to expand.*

  $\mathbf{w}_{k,j+1} \leftarrow \mathbf{w}_{k,j}(\Delta_{k,j})$ ▷ *accept the current trial iterate that sufficiently reduces* θ_2.

  $j \leftarrow j + 1$ ▷ *increment the iteration counter.*

return $\mathbf{w}_{k,j+1}$ ▷ *return the iterate* $\mathbf{w}_{k,j+1}$ *that is acceptable by* $\mathcal{F}$ *and pass condition* (3.61).

where $\sigma^{\text{safe}}_{\min}, \sigma^{\text{safe}}_{\max} \in [0, 1]$ and $\epsilon_2^{\alpha_t} \in (0, 1)$. See Table 3.1.

The safe tangential step procedure sometimes accelerates convergence of the calculation. However, it has been observed that chemical equilibrium calculations in general require fewer iterations if this strategy is disabled.

3.3.4 Convergence Conditions

The algorithm requires a convergence checking scheme in order to detect an optimum iterate. Interior-point methods have the additional requirement that the perturbation parameter $\hat{\mu}$ converges to zero, though in practice we expect $\hat{\mu}$ to become sufficiently small. This is necessary so that the solution of the perturbed problem yields an accurate approximation of the optimum point.

The work of Ulbrich et al. (2004) defines the following optimality error function:

$$e(\mathbf{x}_k, \mathbf{y}_k, \mathbf{z}_k) = \max\left\{ \|\mathbf{c}(\mathbf{x}_k)\|_{\infty}, \frac{\mathbf{x}_k^T \mathbf{z}_k}{s_c}, \frac{\|\nabla_x \mathcal{L}(\mathbf{x}_k, \mathbf{y}_k, \mathbf{z}_k)\|_{\infty}}{s_l} \right\}, \tag{3.66}$$

where subscript k denotes the number of the iteration, and the scaling parameters s_c and s_l are

given by:

$$s_c = \max\left\{1, 0.01\frac{\|\mathbf{z}_k\|_1}{n}\right\} \tag{3.67}$$

and

$$s_l = \max\left\{1, 0.01\frac{\|\mathbf{y}_k\|_1 + \|\mathbf{z}_k\|_1}{m+n}\right\}. \tag{3.68}$$

The previous error function is a slight modification of the one defined in Wächter and Biegler (2005c).

Convergence is assumed whenever:

$$e(\mathbf{w}_k) \leq \epsilon_{\text{tol}}, \tag{3.69}$$

for some small enough $\epsilon_{\text{tol}} > 0$ such as $\epsilon_{\text{tol}} = 10^{-8}$.

3.3.5 Scaling

As discussed in Nocedal and Wright (1999), it is crucial that proper scaling of the variables is done to improve the performance of minimisation algorithms. Indeed, as we shall see later, the primal-dual interior-point algorithm works more efficiently for sequential optimisation calculations if proper scaling of the variables is performed.

Let us denote $\mathbf{D}$ as the diagonal scaling matrix of the primal variable $\mathbf{x}$. Then, we define the scaled primal variable $\bar{\mathbf{x}}$ as:

$$\bar{\mathbf{x}} = \mathbf{D}^{-1}\mathbf{x}, \tag{3.70}$$

and the scaled objective function $\bar{f}(\bar{\mathbf{x}})$ as:

$$\bar{f}(\bar{\mathbf{x}}) = f(\mathbf{x}). \tag{3.71}$$

The scaling of $\mathbf{x}$ has implications on the gradient and Hessian of the scaled objective function $\bar{f}(\bar{\mathbf{x}})$. Let $\bar{\nabla}$ and $\bar{\nabla}^2$ denote the gradient and Hessian operators with respect to $\bar{\mathbf{x}}$. Then, it follows that:

$$\bar{\nabla}\bar{f}(\bar{\mathbf{x}}) = \mathbf{D}\nabla f(\mathbf{x}) \tag{3.72}$$

and

$$\bar{\nabla}^2\bar{f}(\bar{\mathbf{x}}) = \mathbf{D}\nabla^2 f(\mathbf{x})\mathbf{D}. \tag{3.73}$$

When performing sequential equilibrium calculations, the previous solution serves as an excellent option for scaling the primal variables for the next calculation. Therefore, denoting by $\hat{\mathbf{x}}$ the primal solution in the previous calculation, we set:

$$\mathbf{D} = \text{diag}(\hat{\mathbf{x}}). \tag{3.74}$$

Note, however, that for a standalone calculation, or at the beginning of a sequence of optimisation calculations, a good scaling is not available. This follows from the fact that only a poor initial guess can be provided at that time. Therefore, the first optimisation calculation is always performed without scaling.

The performance boost of the calculation with scaling, as we shall see later, is in part explained by the shift of the scaled variables to the interior of the feasible domain. This is a beneficial

condition for interior-point algorithms, which can behave erratically near the boundaries of the feasible domain. Moreover, this solves the issue of some variables having different orders of magnitude, which is very common in chemical equilibrium problems.

3.3.6 Initial Guess

In order to speed up a sequence of optimisation calculations, one can think of using the previous primal and dual solutions $(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}})$ as the initial guess for the next calculation. However, care must be taken, since such approach results in a tiny initial perturbation to the KKT equation (3.18), with $\mu_0 := \mu(\hat{\mathbf{x}}, \hat{\mathbf{z}})$ being sufficiently close to zero. This is because the complementary condition (3.15) was enforced in the previous calculation.

It was observed that the interior-point algorithm of Ulbrich et al. (2004) in general fails when the previous primal and dual solutions are used as initial guess for the next calculation. However, one can remedy this by appropriately modifying $\hat{\mathbf{x}}$ and $\hat{\mathbf{z}}$ so that the resultant initial perturbation parameter μ is not so small.

In order to modify as little as possible the vectors $\hat{\mathbf{x}}$ and $\hat{\mathbf{z}}$, the following strategy for the initial guess is adopted:

$$\mathbf{x}_0 = \max(\hat{\mathbf{x}}, x_0^{\min}) \tag{3.75}$$

$$\mathbf{y}_0 = \hat{\mathbf{y}} \tag{3.76}$$

$$\mathbf{z}_0 = \max(\hat{\mathbf{z}}, z_0^{\min}), \tag{3.77}$$

where the previous operations are component-wise, and the scalar parameters $x_0^{\min}$ and $z_0^{\min}$ are lower-bounds for the entries in $\mathbf{x}_0$ and $\mathbf{z}_0$ respectively (see Table 3.1 for default values).

More restriction should be enforced on $\mathbf{z}_0$ than $\mathbf{x}_0$ by choosing $z_0^{\min}$ larger than $x_0^{\min}$. The idea is to use $z_0^{\min}$ to control the initial perturbation parameter μ, while $x_0^{\min}$ is mainly used to prevent an initial iterate with very small entries. For example, when one phase is not present at equilibrium, the species in that phase will have very tiny number of moles at the end of the calculation.

The previous approach, however, poses a complication. For every minimisation calculation, there is an optimal value for $z_0^{\min}$ that would result in convergence with a minimum number of iterations. Since these values are not known a priori, a trade-off between efficiency and robustness must be made when choosing a single value for $z_0^{\min}$. This value should be small enough to guarantee convergence for all calculations in a reasonable number of iterations.

This complication is avoided with the use of the watchdog strategy described in the next section. As shown there, if the initial perturbation parameter μ is already small enough, then the calculation starts under the watchdog mode. Since in this mode no convergence strategies are performed, the calculation is more insensitive to a small perturbation parameter. Therefore, the previous modification should only be performed if the watchdog strategy is not active.

3.3.7 Watchdog Strategy

The minimisation algorithm of Ulbrich et al. (2004) and Silva et al. (2008) adopts a monotone strategy. In this strategy, every accepted iterate provides a sufficient decrease in either the

optimality or feasibility measures. Its main advantage is to aid convergence from arbitrary initial guesses, since an uncontrolled scheme of the step lengths could result in huge increases of these measures.

Nevertheless, a well-known disadvantage of monotone strategies is that they potentially reject iterates that would make good progress towards the solution (Nocedal and Wright, 1999). This is known as the *Maratos effect*, when the algorithm continuously discards good iterates because they either increase the optimality or feasibility measures. As a result, convergence to a solution becomes very slow.

The Maratos effect can be circumvented with the use of the watchdog technique of Chamberlain et al. (1982). Our adapted approach consists of applying plain Newton's method to the relaxed KKT equations (3.18) with a *constant* perturbation parameter $\dot{\mu}$, rather than the variable $\hat{\mu}$. In addition, normal and tangential components for the trust-region approach are not computed, nor is the new iterate supposed to be acceptable by the filter. Therefore, this technique is considered a non-monotone strategy.

After a sufficient progress has been made in the calculation, our algorithm switches to the watchdog mode. This speeds up convergence towards the optimum solution. The calculation is said to have progressed sufficiently if $\hat{\mu} < \mu_w$, where μ_w is the watchdog threshold, whose default value is $\mu_w = 10^{-2}$. Note, however, that the algorithm leaves a monotone strategy to a non-monotone one, opening up the possibility of divergence. Hence, it is important to monitor the calculation to detect when no progress is being made towards the solution.

The monitoring approach consists of checking after W iterations if the current iterate is acceptable by the filter. If so, then the filter is extended with the current feasibility and optimality measures and a new round of W iterations under the watchdog strategy is allowed. Otherwise, the algorithm reverts to the monotone trust-region strategy, starting from the last watchdog iterate acceptable by the filter. In addition, we monitor the optimality measure ψ_k at every watchdog iteration. The idea is to prevent an uncontrolled increase of that measure. For this we introduce the parameter $\psi_{\max}$ as an upper-bound for ψ_k, so that whenever $\psi_k \geq \psi_{\max}$, the algorithm reverts to the monotone trust-region strategy as before. Table 3.1 presents default values for the previous parameters.

In equation (3.20) Newton's method was applied to the relaxed KKT equations (3.18). An efficient solution of that linear system, assuming a constant perturbation parameter $\dot{\mu}$, can be performed by solving:

$$\bar{\mathbf{J}}(\mathbf{w}) \begin{bmatrix} \Delta\mathbf{x} \\ \Delta\mathbf{y} \end{bmatrix} = - \begin{bmatrix} \nabla_x \mathcal{L}(\mathbf{w}) + \mathbf{z} - \dot{\mu}\mathbf{X}^{-1}\mathbf{e} \\ \mathbf{h}(\mathbf{x}) \end{bmatrix} \tag{3.78}$$

and then, using the previous results to calculate:

$$\Delta\mathbf{z} = -\mathbf{X}^{-1}[\mathbf{Z}\Delta\mathbf{x} + \mathbf{X}\mathbf{z} - \dot{\mu}\mathbf{e}], \tag{3.79}$$

where the reduced Jacobian matrix $\bar{\mathbf{J}}$ is given by equation (3.38). See Table 3.1 for the default value of $\dot{\mu}$.

Recall that the iterates $\mathbf{x}_k$ and $\mathbf{z}_k$ must always stay strictly feasible, which requires $\mathbf{x}_k, \mathbf{z}_k > \mathbf{0}$. Therefore, care must be taken to guarantee that the full step along directions $\Delta\mathbf{x}$ and $\Delta\mathbf{z}$ does

not result in feasibility violations. Let us define the following *fraction-to-the-boundary* steps:

$$\alpha_k^x := \min_i \{\Lambda_i^x \mid \Lambda_i^x \in (0,1]\}, \tag{3.80}$$

$$\alpha_k^z := \min_i \{\Lambda_i^z \mid \Lambda_i^z \in (0,1]\}, \tag{3.81}$$

for $i = 1, \ldots, n$, where Λ_i^x and Λ_i^z are given by:

$$\Lambda_i^x = \Lambda\left(-\frac{\tau x_i}{\Delta x_i}\right) \quad \text{and} \quad \Lambda_i^z = \Lambda\left(-\frac{\tau z_i}{\Delta z_i}\right) \tag{3.82}$$

with operator $\Lambda\colon \mathbb{R} \mapsto \mathbb{R}$ defined by:

$$\Lambda(x) := \begin{cases} x & \text{if } 0 < x < 1 \\ 1 & \text{otherwise} \end{cases}. \tag{3.83}$$

The parameter $\tau = 0.995$ is used to move the new iterates slightly away from the bounds.

Once the step lengths α_k^x and α_k^z have been computed, the following equations are applied to advance the iterates $\mathbf{x}_k$, $\mathbf{y}_k$ and $\mathbf{z}_k$:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k^x \Delta\mathbf{x}, \tag{3.84}$$

$$\mathbf{y}_{k+1} = \mathbf{y}_k + \Delta\mathbf{y}. \tag{3.85}$$

$$\mathbf{z}_{k+1} = \mathbf{z}_k + \alpha_k^z \Delta\mathbf{z}. \tag{3.86}$$

Chapter 5 presents convergence plots of the trust-region interior-point method of Ulbrich et al. (2004) and Silva et al. (2008) with and without the watchdog strategy. The results indicate that the watchdog strategy boost the performance of the calculations with quadratic convergence rates.

3.3.8 Phase Stability Test

Once an equilibrium calculation has been performed, it is important to determine the stability of the phases in the chemical system. The presented interior-point minimisation algorithm alone produces a rough estimate of which phases are unstable, since these phases have a small number of moles with respect to the total in the system. However, this simplistic phase stability test is not very accurate and helpful, since the small number of moles of the unstable phases are very sensitive to the perturbation parameter $\hat{\mu}$ of the interior-point method. In addition, it does not indicate how far from equilibrium the unstable phases are.

Therefore, a phase stability test similar to the one presented by Kulik et al. (2013) is adopted, which has been successfully applied in the geochemical package GEM-Selektor. In this test, stability indices for all phases are readily calculated from the Lagrange multipliers of the minimisation calculation. The advantage of this approach over the analysis of the relative number of moles of the phases is that these indices are less sensitive to algorithmic parameters, providing a more accurate indication of which phases are unstable. In addition, it also provides a quantitative measure of how far they are from equilibrium.

In Kulik et al. (2013), the stability index Λ_π of the π-th phase is defined as:

$$\Lambda_\pi := \log_{10} \Omega_\pi, \tag{3.87}$$

where Ω_π is the generalised saturation index of the same phase. Differently from Kulik et al. (2013), however, Ω_π is defined as:

$$\Omega_\pi := \sum_{i \in \mathcal{I}_\pi} x_{i^\star}^\pi \exp\left(-\frac{z_i}{RT}\right), \tag{3.88}$$

where $\mathcal{I}_\pi$ is the set of indices of the species in the π-th phase; $i^\star$ is the local index of the i-th species in its phase; $x_{i^\star}^\pi$ is the molar fraction of the i-th species; and z_i is the i-th component of the vector of Lagrange multipliers $\mathbf{z}$. Note that the minimisation algorithm of Kulik et al. (2013) does not introduce the additional Lagrange multipliers $\mathbf{z}$, and so the calculation of Ω_π uses a different, but equivalent, equation.

After the calculation of the phase stability indices, the unstable phases can be identified. For every stability index Λ_π such that $|\Lambda_\pi| \geq \epsilon_\Lambda$, the π-th phase is classified as unstable, where ϵ_Λ is the phase stability tolerance, whose default value is $\epsilon_\Lambda = 0.01$. Then, their molar composition is zeroed and the mass-balance of the stable phases is corrected. This procedure ensures that the equilibrium solution does not violate the Gibbs phase rule.

To perform this mass-balance correction procedure, new equilibrium calculations are required. However, at this time, only the stable phases need to be considered. Observe that this correction approach is in general very efficient, because *(i)* a good initial guess is already known, and *(ii)* the size of the chemical system has been reduced. Therefore, using a quadratic convergent equilibrium method should require one or two iterations in general.

The equilibrium method based on the law of mass-action presented in Chapter 2 is used to perform these mass-balance corrections. This is because the linear systems solved in that method have dimension N, instead of $N + C$ in the interior-point algorithm. In addition, it does not involve any Lagrange multipliers, and so the state of the multipliers $\mathbf{y}$ and $\mathbf{z}$ of the interior-point method is not artificially altered, which could compromise subsequent equilibrium calculations that require these as initial guesses.

CHAPTER 4

CHEMICAL KINETICS

This chapter presents the governing equations that model the compositional evolution of a chemical system subject to reactions controlled by kinetics and equilibrium, and proposes a novel methodology for solving them. The formulation assumes a closed-system for simplicity. Addition of source and sink contributions to the equations should be straightforward. Moreover, the kinetic processes are assumed to occur in a well-mixed batch reactor, where transport phenomena such as diffusion, convection and dispersion are neglected. However, the algorithm is designed to be coupled to a transport simulator in future work.

4.1 Introduction

Chemical equilibrium calculations only provide the final thermodynamic state of a system undergoing an isobaric and isothermal process. Thus, it does not offer an insight of how the system evolves with time until its equilibrium state is established. The transient changes in the state of a chemical system can be obtained, nevertheless, by adopting a chemical kinetics model for the reactive processes.

Figure 4.1 illustrates the time evolution of a chemical system composed of an aqueous solution and mineral calcite. The transient state of the system not only provides how aqueous solutes respond to the dissolution of a solid phase, but also the time scale at which the equilibrium state is achieved. Such information is important, for example, to estimate the time required for the injected carbon dioxide in a saline aquifer to equilibrate with both subsurface fluid and rock-forming minerals.

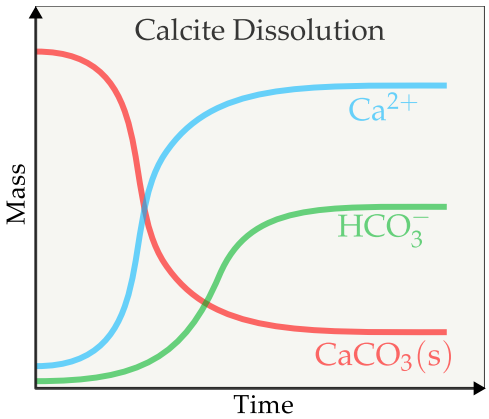


Figure 4.1: Schematic representation of the kinetic dissolution of calcite.

Consider the following chemical reactions

taking place during calcite dissolution:

Aqueous Reactions

$$(r_1) \qquad \mathrm{H_2O(l)} \rightleftharpoons \mathrm{H^+ + OH^-}, \tag{4.1}$$

$$(r_2) \qquad \mathrm{CO_2(aq) + H_2O(l)} \rightleftharpoons \mathrm{HCO_3^- + H^+}, \tag{4.2}$$

Mineral Reactions

$$(r_3) \qquad \mathrm{CaCO_3(s) + H^+} \rightleftharpoons \mathrm{HCO_3^- + Ca^{+2}}, \tag{4.3}$$

where r_1, r_2 and r_3 denote the reaction rates[1] of the reactions, in units of mol/s. Several more reactions and aqueous species may be necessary to model this problem more accurately. However, assume only these reactions and species for now, which will make it simpler to present the governing equations of the reactive process.

Assume that the reaction rates are positive when the reactions proceed from left to right. By accounting for the production and consumption of the species on both sides of the reactions, the following differential equations can be derived:

$$\frac{\mathrm{d}n_{\mathrm{H_2O(l)}}}{\mathrm{d}t} = -r_1(\mathbf{n}) - r_2(\mathbf{n}), \tag{4.4}$$

$$\frac{\mathrm{d}n_{\mathrm{H^+}}}{\mathrm{d}t} = \quad r_1(\mathbf{n}) + r_2(\mathbf{n}) - r_3(\mathbf{n}), \tag{4.5}$$

$$\frac{\mathrm{d}n_{\mathrm{OH^-}}}{\mathrm{d}t} = \quad r_1(\mathbf{n}), \tag{4.6}$$

$$\frac{\mathrm{d}n_{\mathrm{HCO_3^-}}}{\mathrm{d}t} = \quad r_2(\mathbf{n}) + r_3(\mathbf{n}), \tag{4.7}$$

$$\frac{\mathrm{d}n_{\mathrm{CO_2(aq)}}}{\mathrm{d}t} = -r_2(\mathbf{n}), \tag{4.8}$$

$$\frac{\mathrm{d}n_{\mathrm{Ca^{2+}}}}{\mathrm{d}t} = \quad r_3(\mathbf{n}), \tag{4.9}$$

$$\frac{\mathrm{d}n_{\mathrm{CaCO_3(s)}}}{\mathrm{d}t} = -r_3(\mathbf{n}), \tag{4.10}$$

where the reaction rates depend on the molar abundance of the species $\mathbf{n}$. Note, for example, how the consumption of $\mathrm{H_2O(l)}$ in both reactions (1) and (2) reflects on the right-hand side of its governing equation. Therefore, integrating these ordinary differential equations from an initial state $\mathbf{n}_0$ suffices to determine the transient state of the multiphase chemical system.

In the following sections the above approach will be generalised for any multiphase system. In addition to this, it will be shown how the differential equations can be simplified to take into account the fact that most aqueous reactions are considerably faster than mineral reactions.

4.2 Governing Equations

Consider the following linearly independent reactions taking place in a chemical system:

$$0 \rightleftharpoons \sum_{i=1}^{N} \nu_{ji} \alpha_i \qquad (j = 1, \ldots, M), \tag{4.11}$$

[1] For an elementary reaction $0 \rightleftharpoons \sum_i \nu_i \alpha_i$ occurring in a closed system, its reaction rate is defined as $r = \frac{1}{\nu_i} \frac{\mathrm{d}n_i}{\mathrm{d}t}$ for any i such that $\nu_i \neq 0$ (Zhang, 2008).

where α_i is the i-th chemical species; ν_{ji} is the stoichiometric coefficient of the i-th species in the j-th reaction; N is the number of chemical species; and M is the number of reactions. The stoichiometric coefficients in any reaction are assumed to be positive for products and negative for reactants.

From the theory of chemical kinetics, it follows that the compositional evolution of a system is governed by the following ordinary differential equations:

$$\frac{\mathrm{d}n_i}{\mathrm{d}t} = f_i(T, P, \mathbf{n}) \qquad (i = 1, \ldots, N), \tag{4.12}$$

where t is the time variable; n_i is the number of moles of the i-th species; $\mathbf{n} \in \mathbb{R}^N$ is the molar composition vector of the system; T and P are the given temperature and pressure of the system; and $f_i \colon \mathbb{R}^{2+N} \mapsto \mathbb{R}$ is defined by:

$$f_i(T, P, \mathbf{n}) \coloneqq \sum_{j=1}^{M} \nu_{ji} r_j(T, P, \mathbf{n}), \tag{4.13}$$

which accounts for the production and consumption of the i-th species in every reaction (4.11). The kinetic rate function of the j-th reaction is denoted by $r_j \colon \mathbb{R}^{2+N} \mapsto \mathbb{R}$. The convention adopted is that r_j is positive when the reaction proceeds towards the products, and negative towards the reactants.

The system of differential equations (4.12) can be written in matrix notation as:

$$\frac{\mathrm{d}\mathbf{n}}{\mathrm{d}t} = \mathbf{f}(\mathbf{n}), \tag{4.14}$$

with $\mathbf{f} \colon \mathbb{R}^N \mapsto \mathbb{R}^N$ defined by:

$$\mathbf{f}(\mathbf{n}) \coloneqq \boldsymbol{\nu}^T \mathbf{r}(\mathbf{n}) \tag{4.15}$$

where $\boldsymbol{\nu} \in \mathbb{R}^{M \times N}$ denotes the stoichiometric matrix of reactions (4.11); and $\mathbf{r} \colon \mathbb{R}^N \mapsto \mathbb{R}^M$ the rate function of these reactions.

Note that for convenience reasons the dependence of temperature and pressure have been omitted from equations (4.14) and (4.15).

4.2.1 Partial Equilibrium

The reactions in geochemical systems proceed with different speeds. Their time scales can differ from each other by several orders of magnitude, ranging from microseconds to millennia (Lasaga, 1998). Langmuir (1996) presents a list containing some common geochemical reactions and their respective half-times[2]. For example, the homogeneous acid-base reaction involving only solutes:

$$\mathrm{H_2CO_3(aq)} \rightleftharpoons \mathrm{H^+ + HCO_3^-} \tag{4.16}$$

has a half-time of about 10^{-6} seconds. The homogeneous[3] solute-water hydration reaction:

$$\mathrm{CO_2(aq) + H_2O(l)} \rightleftharpoons \mathrm{H_2CO_3(aq)} \tag{4.17}$$

[2]The necessary time to consume half of the initial amount of a reactant.

[3]In a homogeneous reaction, the reactants and products are contained in the same phase, while in a heterogeneous reaction they are contained in different phases.

has a half-time of 0.1 seconds. Compare these time scales with the half-time of the heterogeneous mineral dissolution reaction:

$$CaCO_3(s) + H^+ \rightleftharpoons Ca^{2+} + HCO_3^-, \tag{4.18}$$

which can be at the order of weeks at low temperatures.

In general, heterogeneous reactions are slower than homogeneous reactions. This is because their reaction rates are controlled by the surface area of the involved phases, and also by local transport phenomena such as diffusion. In case of mineral precipitation, the rate is mainly dependent on the mechanisms of crystal nucleation and growth. This is in contrast with homogeneous reactions, where all participating species are constantly colliding with each other in the same phase.

These broad differences in the speed of the reactions pose several numerical complications. The ordinary differential equations (4.14) are severely stiff, requiring appropriate methods for its integration. However, a carefully selected numerical method might still need tiny time steps in order to capture the kinetics of the fastest reaction in the system. It is not optimal to use time steps in the order of microseconds when there are some reactions in the system requiring years to achieve some progress, and where the application of interest has time scales of millennia, such as for carbon storage. Therefore, a simplification is necessary to allow larger time steps for efficient integration and still provide accurate calculations.

It is plausible to assume partial equilibrium in some geochemical processes (Helgeson, 1968; Helgeson et al., 1969, 1970). Consider the dissolution of calcite in an aqueous solution given by reaction (4.18). Recall that the speed of the reactions involving only aqueous solutes are, in general, considerably faster than the speed of this reaction. Thus, it is reasonable to consider that the aqueous solutes are in equilibrium at all times during the process, while calcite is kinetically reacting, and thus out of equilibrium with them. This assumption has also been adopted by Lichtner (1985), Steefel and Cappellen (1990) and Steefel and Lasaga (1994).

The partial equilibrium assumption eliminates the dependence of the calculations on the time scales of the fast reactions. Because only the slow reactions are assumed to be controlled by kinetics, while the fast reactions are controlled by equilibrium, the rate laws of the latter are no longer necessary. Their equilibrium conditions are governed by algebraic constraints instead of differential ones.

In addition, the partial equilibrium assumption simplifies the modelling. Assuming all reactions in geochemical processes to be controlled by kinetics can be a daunting task. For example, the rate law of every reaction would be necessary, which in general requires several temperature and pressure dependent parameters. Moreover, every heterogeneous reaction, including the gaseous-aqueous reactions, would require some reactive surface area model, increasing the complexity of the modelling.

Nevertheless, care must be taken not to assume partial equilibrium inappropriately. Analysing the rates of all reactions occurring in the process is fundamental for identifying the fast and slow reactions and guaranteeing some degree of accuracy. Sometimes, however, accuracy can be compromised by modelling a reaction with equilibrium control.

4.2.2 Partitioning

The reactions are classified in two groups: fast reactions and slow reactions. Following the discussion of partial equilibrium in the previous section, the slow reactions will be assumed to be controlled by kinetics, and the fast reactions to be controlled by equilibrium.

Define a *kinetic reaction* as a reaction controlled by kinetics, and an *equilibrium reaction* as a reaction controlled by equilibrium. In addition, define the following terms:

equilibrium species is a species involved in an equilibrium reaction,

kinetic species is a species involved in a kinetic reaction, but **not** in any equilibrium reaction,

inert species is a species not involved in any reaction,

which will be useful in the formulation of the governing equations of chemical kinetics coupled with chemical equilibrium.

Let $\boldsymbol{\alpha}_e$, $\boldsymbol{\alpha}_k$, and $\boldsymbol{\alpha}_i$ denote the set of equilibrium, kinetic and inert species respectively. Moreover, let N_e, N_k, and N_i denote the respective number of equilibrium, kinetic, and inert species. From the previous definitions, it follows that:

$$\boldsymbol{\alpha}_e \cup \boldsymbol{\alpha}_k \cup \boldsymbol{\alpha}_i = \boldsymbol{\alpha} \tag{4.19}$$

and

$$\boldsymbol{\alpha}_e \cap \boldsymbol{\alpha}_k \cap \boldsymbol{\alpha}_i = \emptyset, \tag{4.20}$$

where $\boldsymbol{\alpha}$ denotes the set of all species in the system. The set of equilibrium species $\boldsymbol{\alpha}_e$ can be constructed by the union of the species participating in the equilibrium reactions. The set of kinetic species $\boldsymbol{\alpha}_k$, on the other hand, can be constructed using:

$$\boldsymbol{\alpha}_k = \boldsymbol{\alpha} - (\boldsymbol{\alpha}_e \cup \boldsymbol{\alpha}_i), \tag{4.21}$$

which can be derived from conditions (4.19) and (4.20).

In order to elucidate the partitioning of the species in equilibrium, kinetic and inert species, consider the example chemical system in Table 4.1. The reactions occurring in this system is listed in Table 4.2. Note that for the modelling of the chemical kinetics of this system, the reactions involving aqueous and gaseous species are assumed to be controlled by equilibrium. Also, due to the fast rates of dissolution and precipitation of mineral halite, its reaction is also assumed to be controlled by equilibrium. The reactions involving calcite, magnesite and dolomite were assumed to be controlled by kinetics, because their rates

Table 4.1: Description of the chemical system H_2O–CO_2–Halite–Calcite–Magnesite–Dolomite with their phases and respective chemical species.

Aqueous Phase	**Gaseous Phase**
$H_2O(l)$	$CO_2(g)$
H^+	$H_2O(g)$
OH^-	**Mineral Phase #1**
HCO_3^-	$NaCl(s)$ *(Halite)*
CO_3^{2-}	**Mineral Phase #2**
Na^+	$CaCO_3(s)$ *(Calcite)*
Cl^-	**Mineral Phase #3**
Ca^{2+}	$MgCO_3(s)$ *(Magnesite)*
Mg^{2+}	**Mineral Phase #4**
$H_2CO_3(aq)$	$(CaMg)(CO_3)_2(s)$ *(Dolomite)*
$CO_2(aq)$	
$CaCO_3(aq)$	
$MgCO_3(aq)$	
$CaCl_2(aq)$	

Table 4.2: Description of the equilibrium and kinetic reactions in the chemical system of Table 4.1.

Equilibrium Reactions	**Kinetic Reactions**
$H_2O(l) \rightleftharpoons H^+ + OH^-$	$CaCO_3(s) + H^+ \rightleftharpoons HCO_3^- + Ca^{+2}$
$CO_2(aq) + H_2O(l) \rightleftharpoons HCO_3^- + H^+$	$MgCO_3(s) + H^+ \rightleftharpoons HCO_3^- + Mg^{+2}$
$H_2CO_3(aq) \rightleftharpoons HCO_3^- + H^+$	$(CaMg)(CO_3)_2(s) + 2H^+ \rightleftharpoons 2HCO_3^- + Ca^{+2} + Mg^{+2}$
$CO_3^{-2} + H^+ \rightleftharpoons HCO_3^-$	
$CaCO_3(aq) + H^+ \rightleftharpoons HCO_3^- + Ca^{+2}$	
$MgCO_3(aq) + H^+ \rightleftharpoons HCO_3^- + Mg^{+2}$	
$CaCl_2(aq) \rightleftharpoons Ca^{+2} + 2Cl^-$	
$CO_2(g) \rightleftharpoons CO_2(aq)$	
$H_2O(g) \rightleftharpoons H_2O(l)$	
$NaCl(s) \rightleftharpoons Na^+ + Cl^-$	

are not as fast as the others. Based on our previous definitions, the equilibrium and kinetic species can be found in Table 4.3. Note that no inert species were assumed.

4.2.3 Revisited Equations

The formulation in Section 4.2 assumed that all reactions were controlled by kinetics. In this section, however, reactions (4.11) will be separated into equilibrium and kinetic reactions as follows:

$$0 \rightleftharpoons \sum_{i=1}^{N} \nu_{ji}^e \alpha_i \qquad (j = 1, \ldots, M_e) \tag{4.22}$$

and

$$0 \rightleftharpoons \sum_{i=1}^{N} \nu_{ji}^k \alpha_i \qquad (j = 1, \ldots, M_k), \tag{4.23}$$

where ν_{ji}^e and ν_{ji}^k are the stoichiometry of the i-th species in the j-th equilibrium and kinetic reactions respectively; and M_e and M_k are the number of equilibrium and kinetic reactions in the system.

As before, let $\boldsymbol{\nu}_e \in \mathbb{R}^{M_e \times N}$ and $\boldsymbol{\nu}_k \in \mathbb{R}^{M_k \times N}$ denote the stoichiometric matrices of the equilibrium and kinetic reactions respectively. From the partitioning discussion in Section 4.2.2, it follows that equilibrium reactions only contain equilibrium species, while kinetic reactions can include both equilibrium and kinetic species. Therefore, we let $\boldsymbol{\nu}_{ke} \in \mathbb{R}^{M_k \times N_e}$ and $\boldsymbol{\nu}_{kk} \in \mathbb{R}^{M_k \times N_k}$ denote the stoichiometric matrices constructed from the columns of $\boldsymbol{\nu}_k$ corresponding to equilibrium and kinetic species respectively.

Let us now formulate the mathematical equations for a general chemical kinetics problem coupled with equilibrium conditions. From equation (4.14), we can write the following governing equations for the evolution of the molar abundance of the kinetic species:

$$\frac{\mathrm{d}\mathbf{n}_k}{\mathrm{d}t} = \mathbf{f}_k(\mathbf{n}). \tag{4.24}$$

with $\mathbf{f}_k \colon \mathbb{R}^N \mapsto \mathbb{R}^{N_k}$ defined by:

$$\mathbf{f}_k(\mathbf{n}) := \boldsymbol{\nu}_{kk}^T \mathbf{r}_k(\mathbf{n}), \tag{4.25}$$

where $\mathbf{r}_k \colon \mathbb{R}^N \mapsto \mathbb{R}^{M_k}$ denotes the rate function of kinetic reactions (4.23).

Although we can write a similar equation for the equilibrium species:

$$\frac{\mathrm{d}\mathbf{n}_e}{\mathrm{d}t} = \mathbf{f}_e(\mathbf{n}), \tag{4.26}$$

we cannot, as before, write an analytical expression for function $\mathbf{f}_e \colon \mathbb{R}^N \mapsto \mathbb{R}^{N_e}$. Because of the equilibrium conditions imposed by the equilibrium reactions, their rates of production and consumption are, in fact, unknowns in the problem.

In order to understand this, consider the dissolution of calcite as an example. At a given time t, suppose $CaCO_3(s)$ is dissolving at a rate of θ mol/s. If all reactions were controlled by kinetics, then it could be said that θ mol/s is the rate of production of Ca^{2+}. However, because Ca^{2+} is an equilibrium species, as soon as it is produced by a kinetic reaction, it instantaneously reacts with other equilibrium species, and so θ mol/s does not represent its actual rate of production.

Therefore, an alternative approach must be used to evolve the molar abundance of the species without requiring the production/consumption rates of the equilibrium species $\mathbf{f}_e$. For this we will rely on the principle of mass conservation, which allows us to state that the mass that leaves or enters the kinetic partition must, respectively, enter or leave the equilibrium partition.

Figure 4.2 illustrates the chemical system of Table 4.1, with its equilibrium and kinetic species. The figure shows the exchange of element atoms among the equilibrium and kinetic partitions. The fact that these atoms are preserved in the system will allow us to calculate the evolution of the molar abundance of the elements in the equilibrium partition. As a result, the composition of the equilibrium species $\mathbf{n}_e$ can be calculated at any time by solving an equilibrium problem using these elemental molar abundances.

Table 4.3: Partition of the chemical system H_2O–CO_2–Halite–Calcite–Magnesite–Dolomite in equilibrium and kinetic species.

Equilibrium Species	Kinetic Species
$H_2O(l)$	$CaCO_3(s)$
H^+	$MgCO_3(s)$
OH^-	$(CaMg)(CO_3)_2(s)$
HCO_3^-	
CO_3^{2-}	
Na^+	
Cl^-	
Ca^{2+}	
Mg^{2+}	
$H_2CO_3(aq)$	
$CO_2(aq)$	
$CaCO_3(aq)$	
$MgCO_3(aq)$	
$CaCl_2(aq)$	
$CO_2(g)$	
$H_2O(g)$	
$NaCl(s)$	

Let $\mathbf{b} \in \mathbb{R}^E$ denote the molar abundance vector of the chemical elements in the system, and $\mathbf{W} \in \mathbb{R}^{E \times N}$ the formula matrix of all species in the chemical system. The formula matrix $\mathbf{W}$ is defined such that its (j, i)-th entry given by w_{ji} denotes the number of atoms of the j-th element in the i-th species. Therefore, it follows that the molar abundance of the elements can be calculated using:

$$\mathbf{b} = \mathbf{W}\mathbf{n}. \tag{4.27}$$

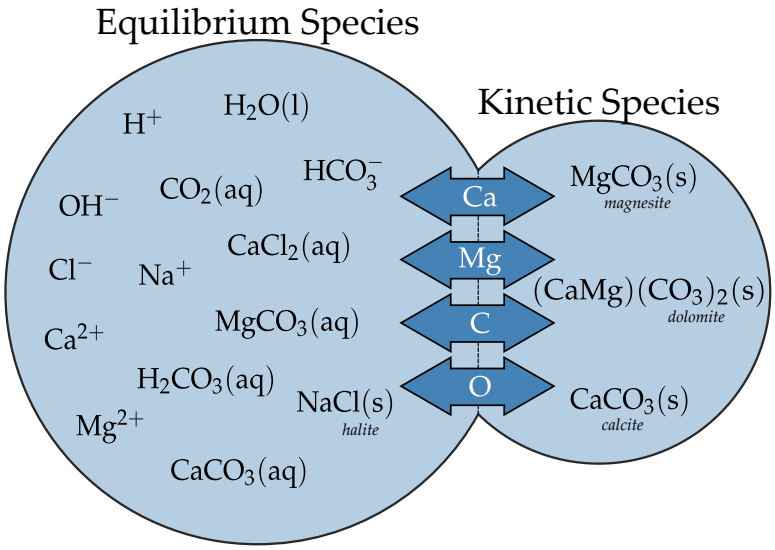


Figure 4.2: Exchange of elemental mass between the equilibrium and kinetic partitions for the chemical system in Table 4.3.

Similarly, we can write the following equations for the equilibrium and kinetic partitions:

$$\mathbf{b}_e = \mathbf{W}_e \mathbf{n}_e \qquad \text{and} \qquad \mathbf{b}_k = \mathbf{W}_k \mathbf{n}_k, \tag{4.28}$$

where $\mathbf{b}_e, \mathbf{b}_k \in \mathbb{R}^E$ are the molar abundance vectors of the chemical elements in the equilibrium and kinetic partitions; and $\mathbf{W}_e \in \mathbb{R}^{E \times N_e}$ and $\mathbf{W}_k \in \mathbb{R}^{E \times N_k}$ are the formula matrices of the equilibrium and kinetic species.

From the principle of mass conservation, it follows that:

$$\frac{\mathrm{d}\mathbf{b}}{\mathrm{d}t} = \frac{\mathrm{d}\mathbf{b}_e}{\mathrm{d}t} + \frac{\mathrm{d}\mathbf{b}_k}{\mathrm{d}t} = \mathbf{0}. \tag{4.29}$$

By multiplying equation (4.24) by $\mathbf{W}_k$ we obtain:

$$\frac{\mathrm{d}\mathbf{b}_k}{\mathrm{d}t} = \mathbf{W}_k \mathbf{f}_k(\mathbf{n}), \tag{4.30}$$

which can be used to write the evolution of the molar abundance of the elements in the equilibrium partition:

$$\frac{\mathrm{d}\mathbf{b}_e}{\mathrm{d}t} = \mathbf{g}_e(\mathbf{n}), \tag{4.31}$$

where $\mathbf{g}_e \colon \mathbb{R}^N \mapsto \mathbb{R}^E$ is defined by:

$$\mathbf{g}_e(\mathbf{n}) := -\mathbf{W}_k \mathbf{f}_k(\mathbf{n}). \tag{4.32}$$

Therefore, combining equations (4.24) and (4.31) we have the following system of ordinary differential equations:

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \mathbf{w}(\mathbf{u}), \tag{4.33}$$

where $\mathbf{u} \colon \mathbb{R}^D \mapsto \mathbb{R}^D$ denotes the unknown function to be integrated; and $\mathbf{w} \colon \mathbb{R}^D \mapsto \mathbb{R}^D$ the right-hand side function of the ordinary differential equation, both defined by:

$$\mathbf{u} := \begin{bmatrix} \mathbf{n}_k \\ \mathbf{b}_e \end{bmatrix} \tag{4.34}$$

and

$$\mathbf{w}(\mathbf{u}) \coloneqq \mathbf{A}\mathbf{r}_k(\mathbf{n}), \tag{4.35}$$

with the coefficient matrix $\mathbf{A} \in \mathbb{R}^{D \times N_k}$ given by:

$$\mathbf{A} \coloneqq \begin{bmatrix} \boldsymbol{\nu}_{kk}^T \\ -\mathbf{W}_k \boldsymbol{\nu}_{kk}^T \end{bmatrix}, \tag{4.36}$$

where $D \coloneqq N_k + E$.

Therefore, equations (4.34) and (4.35) govern the evolution of the molar abundance of both kinetic species $\mathbf{n}_k$ and chemical elements in the equilibrium species $\mathbf{b}_e$. Observe, however, that the rate function $\mathbf{r}_k$ in equation (4.35) depends on the system composition $\mathbf{n}$, which cannot be explicitly obtained from $\mathbf{u}$. In what follows we will see how this problem can be solved.

4.2.4 Chemical Equilibrium

To integrate equations (4.34) and (4.35), it is necessary to construct a function $\boldsymbol{\varphi}\colon \mathbb{R}^D \mapsto \mathbb{R}^N$ such that:

$$\mathbf{n} = \boldsymbol{\varphi}(\mathbf{u}). \tag{4.37}$$

Unfortunately an explicit expression for function $\boldsymbol{\varphi}$ is not available due to the intricate dependence of $\mathbf{n}$ on $\mathbf{u}$. In fact, all the complexity of this dependence lies in the calculation of the molar abundance of the equilibrium species $\mathbf{n}_e$ from the elemental molar abundance $\mathbf{b}_e$. This is because the molar abundance of the kinetic species $\mathbf{n}_k$ can be explicitly obtained from $\mathbf{u} = (\mathbf{u}_1, \mathbf{u}_2)^T = (\mathbf{n}_k, \mathbf{b}_e)^T$ using:

$$\mathbf{n}_k = \mathbf{u}_1. \tag{4.38}$$

Section 4.2.1 introduced the concept of partial equilibrium, and Section 4.2.2 formalised the concept of partitioning the species in a set of equilibrium and kinetic species. As a result, the equilibrium species constitute a sub-system in which chemical equilibrium is always attained.

Therefore, from the principle of minimum Gibbs free energy, the chemical equilibrium state of the equilibrium species can be calculated by solving the following constrained minimisation problem:

$$\min_{\mathbf{n}_e} G_e(\mathbf{n}_e; T, P, \mathbf{n}_k) \quad \text{subject to} \begin{cases} \mathbf{W}_e \mathbf{n}_e = \mathbf{b}_e \\ \mathbf{n}_e \geq \mathbf{0} \end{cases}, \tag{4.39}$$

where $G_e\colon \mathbb{R}^{N_e} \mapsto \mathbb{R}$ denotes the Gibbs free energy function of the equilibrium partition, defined by:

$$G_e(\mathbf{n}_e; T, P, \mathbf{n}_k) \coloneqq \mathbf{n}_e^T \boldsymbol{\mu}_e(\mathbf{n}_e; T, P, \mathbf{n}_k). \tag{4.40}$$

The chemical potential function of the equilibrium species $\boldsymbol{\mu}_e\colon \mathbb{R}^{N_e} \mapsto \mathbb{R}^{N_e}$ is defined by:

$$\boldsymbol{\mu}_e(\mathbf{n}_e; T, P, \mathbf{n}_k) \coloneqq \boldsymbol{\mu}_e^\circ(T, P) + RT \ln \mathbf{a}_e(\mathbf{n}_e; T, P, \mathbf{n}_k), \tag{4.41}$$

where $\boldsymbol{\mu}_e^\circ\colon \mathbb{R}^2 \mapsto \mathbb{R}^{N_e}$ is the standard chemical potential function of the equilibrium species; $\mathbf{a}_e\colon \mathbb{R}^{N_e} \mapsto \mathbb{R}^{N_e}$ is the activity function of the equilibrium species; and R is the universal gas constant.

Remark. The solution of the minimisation problem (4.39), and the evaluation of all functions in equations (4.40) and (4.41) assume T, P and $\mathbf{n}_k$ as constant parameters.

Hence, the function $\boldsymbol{\varphi}$ is defined as the solution of equation (4.38) and the Gibbs energy minimum problem (4.39). This equilibrium problem is solved using the Gibbs energy minimisation algorithm presented in Chapter 3. This method was specifically designed for applications that require sequential equilibrium calculations, such as chemical kinetics and reactive transport modelling.

Performance results of this equilibrium method, as shown later, indicate quadratic rates of convergence near the solution. Therefore, by using the compositional state in a previous time step as the initial guess for the equilibrium calculation in a subsequent step, only a few iterations should be necessary to solve the problem.

4.2.5 Rates of Mineral Reactions

To model the kinetic dissolution and precipitation of minerals, kinetic rate laws for the mineral reactions are necessary. By adapting the mineral rate laws presented in Lasaga (1981); Aagaard and Helgeson (1982); Lasaga (1998); Steefel and Cappellen (1990); Steefel and Lasaga (1994); Perkins et al. (1997); Palandri and Kharaka (2004), the following general rate law for crystal growth and mineral dissolution has been adopted in this work:

$$r_m(T, P, \mathbf{n}) := \mathcal{A}_m(\mathbf{n}) \sum_i \mathcal{M}_{m,i}(T, P, \mathbf{n}) \tag{4.42}$$

where $r_m \colon \mathbb{R}^{2+N} \mapsto \mathbb{R}$ is the rate function of mineral m (in units of moles per unit time); $\mathcal{A}_m$ is the surface area function of the mineral; and $\mathcal{M}_{m,i}$ is the i-th kinetic mechanism function of the mineral (in units of moles per unit surface area and unit time). This functional form of r_m allows us to model several kinetic mineral mechanisms such as acid, neutral, base, carbonate, and so forth (Palandri and Kharaka, 2004).

Estimating the evolution of the mineral surface area in geological formations is very difficult. The process is so intricate that it is still not completely understood even in batch reactors (Zhu, 2009). However, in the interest of understanding the overall kinetic water-rock-gas interactions, a few simplified models have been assumed in the literature.

In Aagaard and Helgeson (1982) the concept of effective surface area was discussed, which is the area of the reactant mineral exposed to the aqueous solution. The effective surface area is, evidently, not necessarily equal to the total surface area of the mineral. Therefore we see that it adds even more complexity to the modelling of mineral surface area, since accounting for the variation of the total surface area is already a challenging task.

Helgeson and Murphy (1983) and Helgeson et al. (1984) adopted a constant total surface for the reactant mineral. It seems to be a more common practice nowadays, however, to adopt models that allow for the variation of the total surface area of the mineral. Two models were implemented for the evolution of mineral surface areas. The simplest one consists of assuming a constant specific surface area σ_m for the mineral, which allows us to compute the surface area by:

$$\mathcal{A}_m := n_m \sigma_m, \tag{4.43}$$

where n_m is the number of moles of mineral m. The other model is based on the one used by PHREEQC (Parkhurst and Appelo, 2013):

$$\mathcal{A}_m \coloneqq \mathcal{A}_m^\circ \left(\frac{n_m}{n_m^\circ}\right)^\eta, \tag{4.44}$$

where $\mathcal{A}_m^\circ$ is the initial surface area of the mineral; n_m° is the initial number of moles of the mineral; and $\eta = 2/3$ for uniformly dissolving cubes and spheres.

In order to model the kinetic dissolution of minerals, it may be necessary to resort to several sources to collect data. In general, each source will present a slightly different equation for the calculation of the mineral rates. As a result, modelling these kinetic processes can be hindered by the necessity to handle a multitude of rate equations.

In Palandri and Kharaka (2004), however, a general and semi-empirical rate equation is presented for the calculation of mineral rates, which is based on the one adopted by GAMSPATH (Perkins et al., 1997). To achieve this uniformity, they analysed mineral dissolution data from several sources, and used them to regress the parameters of the general equation. Therefore, to take advantage of their large mineral kinetic database, the i-th mechanism function $\mathcal{M}_{m,i}$ is defined by:

$$\mathcal{M}_{m,i} \coloneqq \operatorname{sgn}(1-\Omega)\kappa_{m,i}|1-\Omega^{p_i}|^{q_i}\mathcal{C}_{m,i} \tag{4.45}$$

where Ω is the saturation index of the mineral; $\kappa_{m,i}$ is the rate constant of the mineral reaction (in units of moles per unit surface area and unit time); p_i and q_i are empirical exponents used to fit the rate law; and $\mathcal{C}_{m,i}$ is a function to model catalysts and inhibitors of the mineral reaction. The saturation index Ω of the mineral is defined by:

$$\Omega \coloneqq \frac{Q_m}{K_m}, \tag{4.46}$$

where, if the mineral reaction is written as:

$$0 \rightleftharpoons \sum_{i=1}^{N} \nu_i \alpha_i, \tag{4.47}$$

then K_m is its equilibrium constant; and Q_m is its reaction quotient, defined by:

$$Q_m \coloneqq \prod_{i=1}^{N} a_i^{\nu_i}. \tag{4.48}$$

The reaction rate constant $\kappa_{m,i}$ in equation (4.45) depends on temperature. This dependence can be modelled via the Arrhenius equation (Lasaga, 1998) as:

$$\kappa_{m,i} \coloneqq \kappa_{m,i}^\circ \exp\left[-\frac{E_{m,i}}{R}\left(\frac{1}{T}-\frac{1}{298.15}\right)\right], \tag{4.49}$$

where $\kappa_{m,i}^\circ$ is the reaction rate constant at 298.15 K; $E_{m,i}$ is the activation energy; R is the universal gas constant; and T is temperature in K.

Finally, the catalyst/inhibitor function $\mathcal{C}_{m,i}$ is defined as:

$$\mathcal{C}_{m,i} \coloneqq \prod_j a_j^{\xi_j} \prod_g P_g^{\eta_g}, \tag{4.50}$$

where a_j is the activity of the j-th species; P_g is the partial pressure of the g-th gaseous species; and ξ_j and η_g are the exponents of the catalysts, when positive, and inhibitors, when negative. Thus, acid mechanisms can be modelled by setting a non-zero value for $\xi_{\mathrm{H^+}}$, while a carbonate mechanism would require $\eta_{\mathrm{CO_2(g)}}$ to be non-zero. A neutral mechanism, on the other hand, can be modelled by setting both ξ_j and η_g to zero for all species.

The parameters $\kappa^{\circ}_{m,i}$, $E_{m,i}$, ξ_j, and η_g for several minerals can be found in Palandri and Kharaka (2004).

4.3 Numerical Method

Several numerical methods exist in the literature for the integration of equation (4.33). Ascher and Petzold (1998), Hairer et al. (2008) and Hairer and Wanner (2010) present methods for stiff and non-stiff system of ordinary differential equations. From our discussions in the previous sections, however, a suitable method for stiff equations should be adopted because of the large differences that can exist in the speeds of the kinetic reactions. Therefore, an implicit multistep backward differentiation formula (BDF) algorithm is adopted, which is highly efficient and robust for stiff ODEs.

This work uses the package CVODE (Cohen and Hindmarsh, 1996; Hindmarsh et al., 2005) for integration of the chemical kinetics equations (4.33). This solver is based on the well-known algorithm VODE (Brown et al., 1989), with improved interface and added capability for dense and banded matrices, using direct or iterative methods for linear systems. The algorithm uses a Adams–Moulton method for non-stiff ODEs, and a BDF method for stiff ones.

An adaptive control scheme of the time step is adopted in the integration. This ensures small steps in steeper regions and large steps in smoother regions. As a result, both accuracy and efficiency are achieved throughout the calculation. This adaptive control is essential in geochemistry, since in general minerals react very fast initially (a steep region), and then proceed very slowly (a smooth region) until equilibrium. If a constant time step is adopted, then it must be small enough to guarantee that the integration is accurate and stable at the beginning of the process. However, the required initial time step (e.g., 10^{-4} s) is usually orders of magnitude smaller than what should be used near equilibrium (e.g., 10^2 days).

The solver CVODE calculates the solution at a new time step using an implicit scheme, requiring a system of non-linear algebraic equations to be solved. Two approaches are offered for solving these non-linear equations: *Newton iteration* and *functional iteration*. The former uses Newton's method to solve the equations, while the latter uses a successive-substitution method. As a result, the functional iteration approach is advised to be used only for non-stiff ODEs (Hindmarsh et al., 2005).

The Newton iteration approach is adopted in this work. Although it is more suitable for stiff ODEs, resulting in more efficient calculations that use larger time steps, this approach increases the level of complexity of the numerical integration. This is because using Newton's method requires the Jacobian function $\mathbf{J}\colon \mathbb{R}^D \mapsto \mathbb{R}^{D\times D}$ defined by:

$$\mathbf{J}(\mathbf{u}) \coloneqq \frac{\partial \mathbf{w}(\mathbf{u})}{\partial \mathbf{u}}. \tag{4.51}$$

In addition, analytical derivatives of the rate laws and species activities are required in the calculation of the Jacobian matrix. Although the computation of these derivatives requires some computational effort, it allows the use of larger time steps and increases the stability of the integration (Ascher and Petzold, 1998; Hairer and Wanner, 2010).

4.3.1 Jacobian Function

Let us now present a methodology for the calculation of the Jacobian function $\mathbf{J}(\mathbf{u})$. Combining equations (4.35) and (4.51), and applying the chain rule in the derivative term, results in:

$$\mathbf{J} = \mathbf{A}\frac{\partial \mathbf{r}_k}{\partial \mathbf{n}}\frac{\partial \mathbf{n}}{\partial \mathbf{u}}. \tag{4.52}$$

The partial molar derivatives of the reaction rates $\partial \mathbf{r}_k/\partial \mathbf{n}$ can be obtained by differentiating the rate functions either analytically or numerically. This work adopts an analytical approach.

The calculation of the partial derivatives $\partial \mathbf{n}/\partial \mathbf{u}$ is slightly more complicated. From the definition of $\mathbf{u}$ in equation (4.34), it follows that:

$$\begin{aligned}\frac{\partial \mathbf{n}}{\partial \mathbf{u}} &= \begin{bmatrix} \dfrac{\partial \mathbf{n}}{\partial \mathbf{n}_k} & \dfrac{\partial \mathbf{n}}{\partial \mathbf{b}_e} \end{bmatrix} \\ &= \begin{bmatrix} \dfrac{\partial \mathbf{n}}{\partial \mathbf{n}_k} & \dfrac{\partial \mathbf{n}}{\partial \mathbf{n}_e}\dfrac{\partial \mathbf{n}_e}{\partial \mathbf{b}_e} \end{bmatrix},\end{aligned} \tag{4.53}$$

where $\partial \mathbf{n}/\partial \mathbf{n}_e$ and $\partial \mathbf{n}/\partial \mathbf{n}_k$ are constant matrices obtained by extracting the columns of the identity matrix $\mathbf{I} \in \mathbb{R}^{N\times N}$ corresponding to the equilibrium and kinetic species respectively. The matrix $\partial \mathbf{n}_e/\partial \mathbf{b}_e$, on the other hand, needs more effort to be calculated, since it depends on the equations governing the equilibrium state of the equilibrium species.

To calculate $\partial \mathbf{n}_e/\partial \mathbf{b}_e$, we write the Lagrange function $\mathcal{L}$ of the minimisation problem (4.39):

$$\mathcal{L}(\mathbf{n}_e, \mathbf{y}_e, \mathbf{z}_e) := \mathbf{n}_e^T \boldsymbol{\mu}_e(\mathbf{n}_e) + (\mathbf{W}_e\mathbf{n}_e - \mathbf{b}_e)^T\mathbf{y}_e - \mathbf{n}_e^T\mathbf{z}_e, \tag{4.54}$$

where $\mathbf{y}_e \in \mathbb{R}^E$ and $\mathbf{z}_e \in \mathbb{R}^{N_e}$ are Lagrange multipliers. In addition, we write the gradient of the Lagrange function with respect to the molar abundance of the equilibrium species $\mathbf{n}_e$:

$$\nabla_n \mathcal{L}(\mathbf{n}_e, \mathbf{y}_e, \mathbf{z}_e) = \boldsymbol{\mu}_e(\mathbf{n}_e) + \mathbf{W}_e^T\mathbf{y}_e - \mathbf{z}_e. \tag{4.55}$$

Assume that $(\mathbf{n}_e, \mathbf{y}_e, \mathbf{z}_e)$ is the solution of the minimisation problem (4.39). From optimisation theory (Nocedal and Wright, 1999), it follows that the following first-order optimality conditions are satisfied at $(\mathbf{n}_e, \mathbf{y}_e, \mathbf{z}_e)$:

$$\boldsymbol{\mu}_e(\mathbf{n}_e) + \mathbf{W}_e^T\mathbf{y}_e - \mathbf{z}_e = \mathbf{0}, \tag{4.56}$$

$$\mathbf{W}_e\mathbf{n}_e - \mathbf{b}_e = \mathbf{0}, \tag{4.57}$$

$$\mathbf{N}_e\mathbf{z}_e = \mathbf{0}, \tag{4.58}$$

$$\mathbf{n}_e, \mathbf{z}_e \geq \mathbf{0}, \tag{4.59}$$

where $\mathbf{N}_e := \operatorname{diag}(\mathbf{n}_e)$. Applying the derivative operator $\partial/\partial \mathbf{b}_e$ in equations (4.56)–(4.58) yields:

$$\nabla \boldsymbol{\mu}_e(\mathbf{n}_e)\frac{\partial \mathbf{n}_e}{\partial \mathbf{b}_e} + \mathbf{W}_e^T\frac{\partial \mathbf{y}_e}{\partial \mathbf{b}_e} - \frac{\partial \mathbf{z}_e}{\partial \mathbf{b}_e} = \mathbf{0}, \tag{4.60}$$

$$\mathbf{W}_e\frac{\partial \mathbf{n}_e}{\partial \mathbf{b}_e} - \mathbf{I}_e = \mathbf{0}, \tag{4.61}$$

$$\mathbf{Z}_e\frac{\partial \mathbf{n}_e}{\partial \mathbf{b}_e} + \mathbf{N}_e\frac{\partial \mathbf{z}_e}{\partial \mathbf{b}_e} = \mathbf{0}, \tag{4.62}$$

which can be simplified to:

$$(\nabla \boldsymbol{\mu}_e + \mathbf{N}_e^{-1}\mathbf{Z}_e)\frac{\partial \mathbf{n}_e}{\partial \mathbf{b}_e} + \mathbf{W}_e^T \frac{\partial \mathbf{y}_e}{\partial \mathbf{b}_e} = \mathbf{0}, \tag{4.63}$$

$$\mathbf{W}_e \frac{\partial \mathbf{n}_e}{\partial \mathbf{b}_e} = \mathbf{I}_e, \tag{4.64}$$

where $\mathbf{I}_e \in \mathbb{R}^{E\times E}$ is an identity matrix and $\mathbf{Z}_e \coloneqq \operatorname{diag}(\mathbf{z}_e)$.

In order to simplify equations (4.63) and (4.64) even further, let $\mathbf{K}_e$ denote a matrix whose columns form a basis of the kernel of $\mathbf{W}_e$, such that:

$$\mathbf{W}_e\mathbf{K}_e = \mathbf{0}. \tag{4.65}$$

Thus, multiplying equation (4.63) by $\mathbf{K}_e^T$ yields:

$$\mathbf{A}_e \frac{\partial \mathbf{n}_e}{\partial \mathbf{b}_e} = \mathbf{B}_e, \tag{4.66}$$

where matrices $\mathbf{A}_e$ and $\mathbf{B}_e$ are defined as:

$$\mathbf{A}_e \coloneqq \begin{bmatrix} \mathbf{K}_e^T(\nabla \boldsymbol{\mu}_e + \mathbf{N}_e^{-1}\mathbf{Z}_e) \\ \mathbf{W}_e \end{bmatrix} \tag{4.67}$$

and

$$\mathbf{B}_e \coloneqq \begin{bmatrix} \mathbf{0} \\ \mathbf{I}_e \end{bmatrix}. \tag{4.68}$$

Therefore, $\partial \mathbf{n}_e / \partial \mathbf{b}_e$ can be calculated by solving the general system of linear equations (4.66), whose coefficient matrix $\mathbf{A}_e$ can be computed once the equilibrium state of the system has been found (i.e., once $\mathbf{n}_e$, $\mathbf{y}_e$, and $\mathbf{z}_e$ has been calculated). Note that the kernel matrix $\mathbf{K}_e$ should be computed only once in the beginning of the integration for efficiency reasons.

CHAPTER 5

APPLICATIONS: CHEMICAL EQUILIBRIUM

This chapter uses the chemical equilibrium algorithms presented in Chapters 2 and 3 to model problems relevant to carbon dioxide sequestration in saline aquifers. In addition, the efficiency of the Gibbs energy minimisation method will be demonstrated, potentially allowing its integration into critical applications such as reactive transport modelling.

The activity and fugacity coefficient models below have been implemented and are used to perform the calculations. These thermodynamic models are presented in Appendix B.

- For ionic species and solvent water the following models are available:
  - the HKF extended Debye-Hückel activity coefficient model (Helgeson and Kirkham, 1974a,b, 1976; Helgeson et al., 1981); and
  - the HMW Pitzer activity coefficient model (Harvie et al., 1984).
- The Setschenow activity coefficient model for neutral aqueous species other than CO_2(aq).
- The activity coefficient models of Drummond (1981), Duan and Sun (2003), and Harvie et al. (1984) for CO_2(aq).
- The fugacity coefficient models of Spycher and Reed (1988), Spycher et al. (2003), Duan et al. (2006) and Peng and Robinson (1976) for CO_2(g) and H_2O(g).
- The ideal activity coefficient model for mineral species.

Note that, with exception of the fugacity coefficient model of Spycher and Reed (1988), the models of Spycher et al. (2003) and Duan et al. (2006) assume the gaseous phase as an ideal mixture. The implication of this is that the fugacity coefficient of each gas does not depend on phase composition, but only on temperature and pressure.

In our calculations the ideal fugacity coefficient model for H_2O(g) is used whenever the model of Duan et al. (2006) is employed for CO_2(g). This follows from the lack of a non-iterative and

efficient equation for the fugacity coefficient of H_2O(g) in the work of Duan et al. (2006). An alternative would be the use of the non-linear equations of state of Duan et al. (1992), but these are prohibitively expensive to use in a reactive reservoir simulator. Note, however, that this assumption has little impact on the accuracy of our predictions of the solubility of CO_2.

The chemical potential of the species were obtained using the equations of state of Helgeson and Kirkham (1974a); Helgeson et al. (1978); Tanger and Helgeson (1988); Shock and Helgeson (1988) and Shock et al. (1992). The parameters of these equations were taken from the database file of the software package SUPCRT, (Johnson et al., 1992). Moreover, the density of water and its temperature and pressure derivatives, required for the calculation of the chemical potential of aqueous species, were calculated using the equation of state of Wagner and Pruss (2002).

The chemical potential of the species can be used to compute equilibrium constants of reactions whenever necessary. This computation can be skipped by using equilibrium constants available in databases such as those from geochemical packages EQ3/6 (Wolery, 1992a) or PHREEQC (Parkhurst and Appelo, 1999, 2013). However, these databases have constants that are in general only temperature dependent, evaluated at a fixed pressure or at pressures corresponding to the vapour pressure of water. Therefore, at higher pressures, these data are no longer capable of yielding accurate equilibrium results.

5.1 Mutual Solubility of CO_2 and H_2O in Brines

The capability to model the solubility of CO_2 in brines accurately is of utmost importance for understanding CO_2 storage in saline aquifers. This is because most of the injected CO_2 will eventually dissolve in the aqueous phase. Moreover, this dissolution will also acidify the aqueous phase, promoting geochemical reactions between fluid and rock. Therefore, correct predictions of these phenomena will depend on the accuracy of the calculated solubility of carbon dioxide in brines.

There are other aspects that should be considered when modelling the solubility of carbon dioxide in brines. One of them concerns the solubility of H_2O in the CO_2-rich phase. This is because near the injection well a considerable amount of water is evaporated into the injected supercritical CO_2. Since this evaporation is directly related to the precipitation of salts, which decreases the local porosity and permeability of the rock, it can be argued that a certain degree of accuracy in the solubility calculation of vapour water in the CO_2-rich phase is required.

Other important aspects include the salt composition of the brine and its salinity. Frequently, brines contain significant concentrations of salts other than NaCl, such as $CaCl_2$, $MgCl_2$, KCl and so forth. As a consequence, a comprehensive accounting of these salts should not be neglected as this could lead to inaccurate results for the calculated CO_2 solubility.

The calculations presented later in this section will cover these cases.

5.1.1 Modelling Problem

The calculation of carbon dioxide solubility in brines can be obtained by using a chemical equilibrium method. For this, a multiphase system consisting of an aqueous and a gaseous phase must be defined. For example, calculating the solubility of CO_2 in $CaCl_2$ brines can be

achieved by defining the chemical system as shown in Table 5.1. For more complex brines, involving a variety of salts, the system would need to be modified accordingly.

Fortunately, this is an easy computational task if a database containing a list of aqueous, gaseous, and mineral species are available. Determining the relevant species is just a matter of identifying those that contain the same chemical elements: the elements H, O, C, Ca, Cl for the H_2O–CO_2–$CaCl_2$ system. Sometimes, however, the modeller knows beforehand that some classes of species in a database are irrelevant to the problem, and need to be discarded. Hydrocarbon species, for instance, are not necessary for this solubility modelling problem.

Table 5.1: Description of the chemical system H_2O–CO_2–$CaCl_2$ with their phases and respective species.

Aqueous Phase	**Gaseous Phase**
H_2O(l)	CO_2(g)
H^+	H_2O(g)
OH^-	
HCO_3^-	
Ca^{2+}	
Na^+	
CO_3^{2-}	
$CaHCO_3^+$	
$CaCl^+$	
$CaOH^+$	
CO_2(aq)	
$CaCO_3$(aq)	
$CaCl_2$(aq)	

The solubility calculations were performed by assuming a mixture containing 1 kg of H_2O, 10 moles of CO_2, and a variable amount of salt to determine the salting-out effect on the solubility of carbon dioxide. The arbitrary amount of CO_2 is explained by the fact that we want enough gas to saturate the aqueous solution and form a gaseous phase at all considered temperatures, pressures and salinities. This is necessary since our goal is to calculate the saturated solubility of that gas in brine.

Once the amounts of H_2O, CO_2 and the salt (e.g., NaCl, KCl, $CaCl_2$, $MgCl_2$) are specified, one can easily calculate the number of moles of the elements H, O, C, Ca, and Cl. This can then be used as input for a chemical equilibrium calculation. From the obtained solution, the solubility of CO_2 is given by the molality of element C in the aqueous phase, which accounts for all aqueous species containing carbon.

5.1.2 Salinity Effect on Solubility

Figures 5.1–5.2 show the solubility of CO_2 in pure water and the solubility of H_2O in the CO_2-rich phase respectively, in terms of the molar fractions x_{CO_2} and y_{H_2O}.

$$x_{CO_2} = \frac{n_{HCO_3^-} + n_{CO_2(aq)}}{n_t^a} \tag{5.1}$$

and

$$y_{H_2O} = \frac{n_{H_2O(g)}}{n_t^g}, \tag{5.2}$$

where n_t^a and n_t^g are the total number of moles in the aqueous and gaseous phase respectively. The experimental solubility data compiled in Spycher et al. (2003) were used to assess the accuracy of the calculations. The fugacity coefficient model of Spycher et al. (2003) was used for the gaseous species CO_2(g) and H_2O(g). Note that the CO_2-rich phase exists in the liquid, gaseous or supercritical state depending on the temperature and pressure of the system.

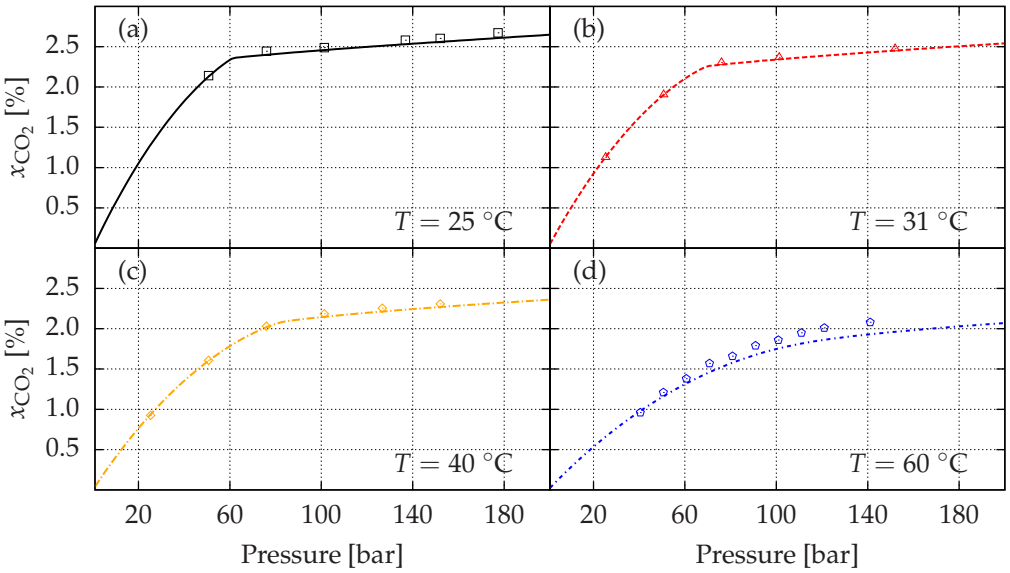


Figure 5.1: Comparison of our calculations of CO_2 solubility in pure water (lines), using the activity coefficient model of Drummond (1981) for CO_2(aq) and the fugacity coefficient model of Spycher et al. (2003) for CO_2(g) and H_2O(g), with the experimental solubility data compiled in Spycher et al. (2003) (points). The calculations assumed a H_2O–CO_2 system composed of an aqueous and gaseous phase. Temperatures used were (a) 25 °C; (b) 31 °C; (c) 40 °C; and (d) 60 °C.

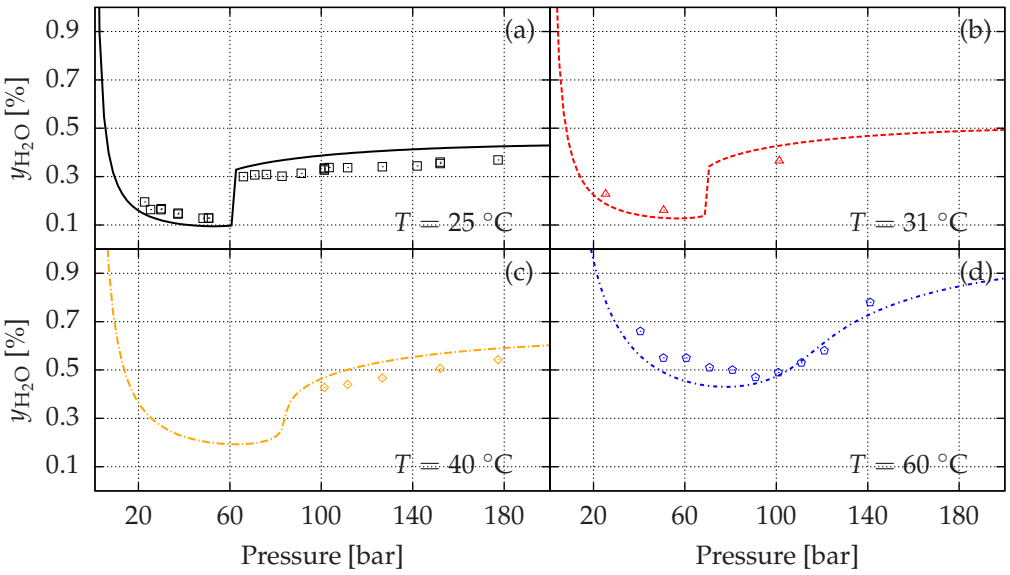


Figure 5.2: Comparison of our calculations of H_2O solubility in the CO_2-rich phase (lines), using the activity coefficient model of Drummond (1981) for CO_2(aq) and the fugacity coefficient model of Spycher et al. (2003) for CO_2(g) and H_2O(g), with the experimental solubility data compiled in Spycher et al. (2003) (points). The calculations assumed a H_2O–CO_2 system composed of an aqueous and gaseous phase. Temperatures used were (a) 25 °C; (b) 31 °C; (c) 40 °C; and (d) 60 °C.

Figure 5.3 shows the solubility of CO_2 in NaCl brines of 0, 1, 2 and 4 molal, at temperatures 30, 60, 90 °C and pressures 1–500 bar. Figure 5.4 shows the corresponding H_2O solubility in the CO_2-rich phase, where no comparisons with experimental data were made because of the lack of these at the assumed temperatures. These equilibrium calculations used the activity coefficient model of Drummond (1981) for CO_2(aq) and the fugacity coefficient model of Spycher et al. (2003) for CO_2(g) and H_2O(g). The calculations in Figure 5.3 were compared against the calculated solubility from the Duan and Sun (2003) model.

These results show that the solubility of carbon dioxide decreases with an increase in salinity: the so called *salting-out effect*. Therefore, one could argue that very saline aquifers could be unfavourable for carbon dioxide storage, since they have a lower potential to trap gases as dissolved carbonate species. However, these solubility results can be misleading, since they do not take into account the water-gas-rock effects in the reservoir, which we consider later.

Sometimes complex water-gas-rock effects increase the potential of the subsurface fluid to dissolve more carbon dioxide. As shown later in Chapter 6, this is what happens in a carbonate aquifer, where it is observed that the dissolution of the carbonate minerals promotes further dissolution of mobile supercritical CO_2. Thus, we see that the storage assessment of carbon storage in saline aquifers should take into account the relevant subsurface chemical processes.

5.1.3 Accuracy Assessment of the Calculations

Tables 5.2–5.7 present the solubility results for carbon dioxide in brines of different salt composition and salinity, at temperature and pressure conditions representative of saline aquifers. The experimental solubility data used to assess the accuracy of the calculations were taken from the measurements of Hou et al. (2013a,b) and Tong et al. (2013).

Table 5.2 presents the results of the solubility calculations in pure water. The equilibrium calculations were performed using different fugacity coefficient models for the gaseous mixture H_2O–CO_2. The results indicate that the fugacity coefficient model of Duan et al. (2006) yields the most accurate results, where an average deviation of 6.3% from the experimental data can be observed.

Tables 5.3–5.4 show the calculated solubility of CO_2 in NaCl brines at high salinities using, respectively, the activity coefficient models of Duan and Sun (2003) and Drummond (1981) for the aqueous species CO_2(aq). Between these two activity coefficient models, it can be seen that the latter is slightly more accurate for NaCl brines. As to the fugacity coefficient models for CO_2(g), we observe again that the model of Duan et al. (2006) yields the most accurate results, where the deviations are in general less than 7%. This good agreement of the results demonstrate that this model is sufficiently accurate for engineering applications.

In order to have a qualitative perception of the accuracy of the fugacity coefficient models of Spycher and Reed (1988), Spycher et al. (2003) and Duan et al. (2006), the reader is referred to Figures 5.5–5.6. The qualitative comparison of the activity coefficient models of Duan and Sun (2003) and Drummond (1981) can be made in Figures 5.7–5.8. Observing these figures one realises that the fugacity coefficient of Duan et al. (2006) for CO_2(g) results in more accurate solubility calculations. Moreover, the activity coefficient model of Drummond (1981) for CO_2(aq) performs better than the one of Duan and Sun (2003) for most of the studied range of

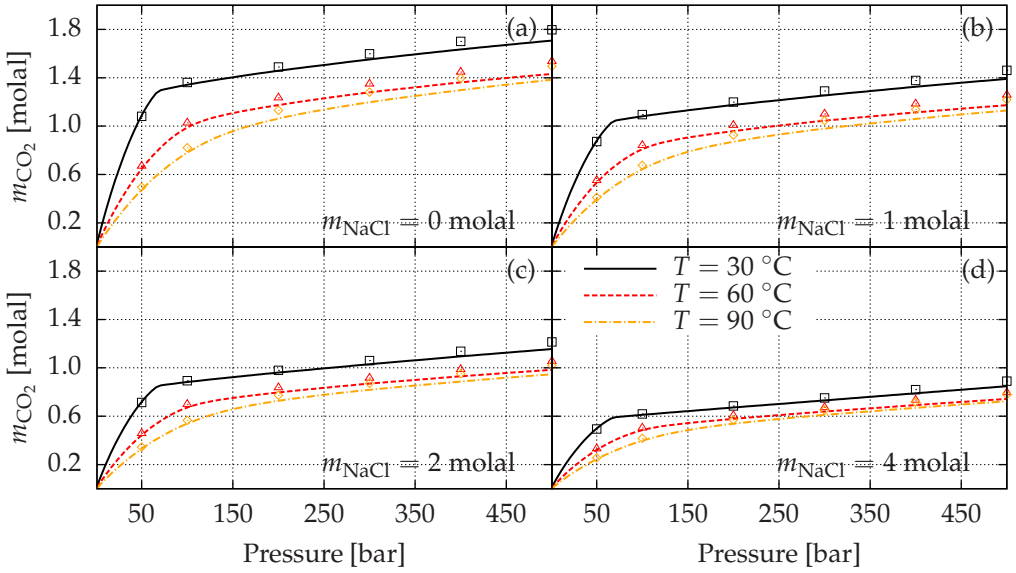


Figure 5.3: Comparison of our calculations of CO_2 solubility in NaCl brine (lines), using the activity coefficient model of Drummond (1981) for CO_2(aq) and the fugacity coefficient model of Spycher et al. (2003) for CO_2(g) and H_2O(g), with the calculated solubility data of Duan and Sun (2003) (points). The calculations assumed a CO_2-H_2O-NaCl system composed of an aqueous and gaseous phase.

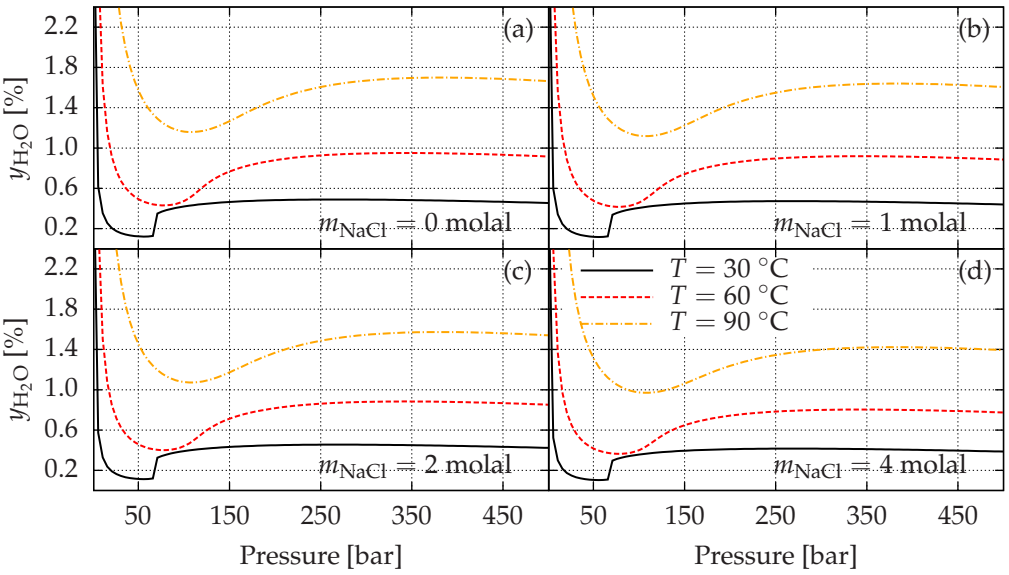


Figure 5.4: Calculated H_2O solubility in the CO_2-rich phase, using the activity coefficient model of Drummond (1981) for CO_2(aq) and the fugacity coefficient model of Spycher et al. (2003) for CO_2(g) and H_2O(g). The calculations assumed a CO_2-H_2O-NaCl system composed of an aqueous and gaseous phase.

temperatures, pressures and salinities.

Tables 5.5–5.7 indicate that the activity coefficient model of Drummond (1981) for CO_2(aq) is not suitable for brines with salt compositions other than pure NaCl. The solubility calculations in NaCl+KCl brines, with KCl salt in low proportion, show that the Duan and Sun (2003) model is slightly superior. In addition, for the solubility calculations in $MgCl_2$ and $CaCl_2$ brines, it becomes evident that the Duan and Sun (2003) model is considerably superior than the Drummond (1981) model for non-NaCl brines.

In Spycher and Pruess (2005), the solubility of H_2O in the CO_2-rich phase was calculated for CO_2-brine system. However, at that time no experimental data were available for assessing the accuracy of their calculations. In Figures 5.9–5.10 we show a comparison of our calculations of H_2O solubility in the CO_2-rich phase with the experimental solubility data of Hou et al. (2013b). Note that this comparison is done in terms of the molar fraction of CO_2(g). These were performed assuming an aqueous phase containing 2.5 and 4.0 molal of NaCl. Different fugacity coefficient models were used for the gaseous species CO_2(g) and H_2O(g). From these figures, it can be observed that the fugacity coefficient model of Duan et al. (2006) achieves better accuracy than the other models, except at higher temperatures, where the model of Spycher et al. (2003) presents a closer agreement with the experimental data. However, we remark that this model was developed for lower temperatures 12–100 °C, as stated in Spycher et al. (2003).

5.2 Phase Behaviour of H_2O–CO_2–Rock Systems

Figure 5.11 shows the ability of the chemical equilibrium method to determine the stable phase assemblage of H_2O–CO_2–mineral systems, where the minerals assumed were halite, calcite, and magnesite respectively. The calculations were made by assuming a fixed feed molar fraction z_{Min} of each mineral component. The feed molar fraction z_{CO_2} of component CO_2 was varied from 0 to $1 - z_{\text{min}}$ and the feed molar fraction z_{H_2O} of component H_2O was computed by:

$$z_{H_2O} = 1 - z_{CO_2} - z_{\text{Min}}. \tag{5.3}$$

From the molar fractions of the components, the molar abundance of each element was determined, which served as input to the equilibrium calculation.

It can be seen in Figure 5.11 that some discontinuities in the concentration of the aqueous species. These discontinuities characterise a change in the phase assemblage of the system. When z_{CO_2} assumes small values, only the aqueous phase exists in the system, which is represented by the steep concentration line of CO_2(aq) on the left side of the graphs. Then, a gaseous phase is formed for intermediate values of z_{CO_2}, while the mineral is still completely dissolved in the aqueous phase. As z_{CO_2} further increases, a solid phase appears in the system, represented by the planar region on the right side of the graphs. Finally, for values of z_{CO_2} near to its limit, we have the aqueous phase being fully evaporated to the gaseous phase, resulting in zero concentrations for the aqueous species.

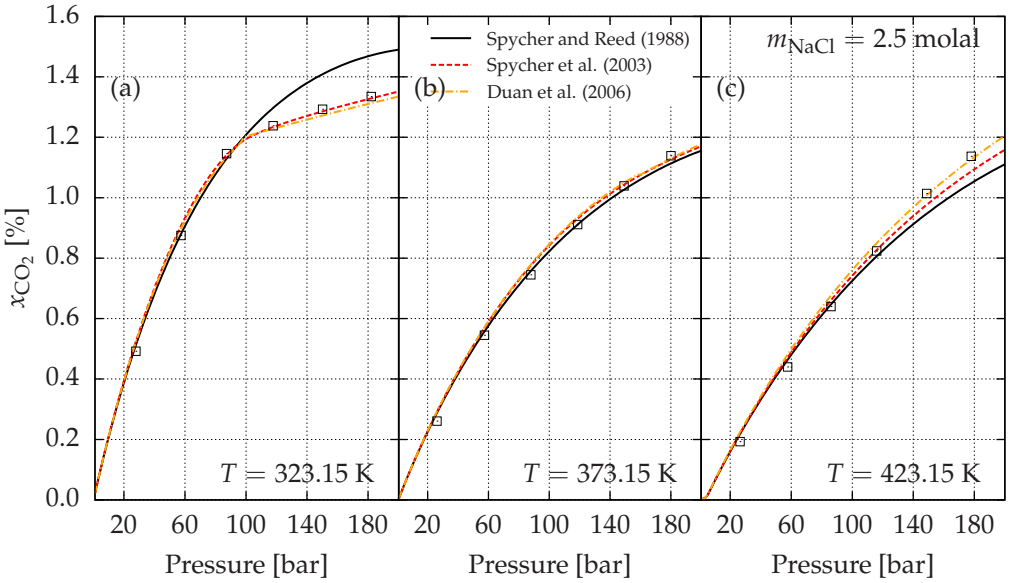


Figure 5.5: Comparison of our calculations of CO_2 solubility in 2.5 molal NaCl brine (lines), using the activity coefficient model of Duan and Sun (2003) for CO_2(aq) and the fugacity coefficient models of Spycher and Reed (1988), Spycher et al. (2003) and Duan et al. (2006) for CO_2(g) and H_2O(g), with the experimental solubility results of Hou et al. (2013b) (points). The calculations assumed a H_2O–CO_2–NaCl system composed of an aqueous and gaseous phase.

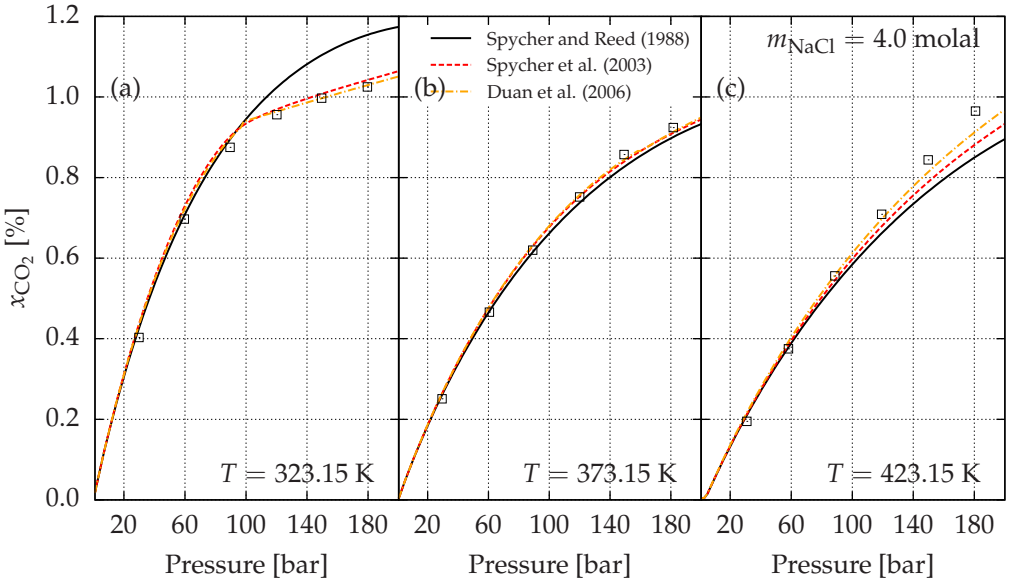


Figure 5.6: Comparison of our calculations of CO_2 solubility in 4.0 molal NaCl brine (lines), using the activity coefficient model of Duan and Sun (2003) for CO_2(aq) and the fugacity coefficient models of Spycher and Reed (1988), Spycher et al. (2003) and Duan et al. (2006) for CO_2(g) and H_2O(g), with the experimental solubility results of Hou et al. (2013b) (points). The calculations assumed a H_2O–CO_2–NaCl system composed of an aqueous and gaseous phase.

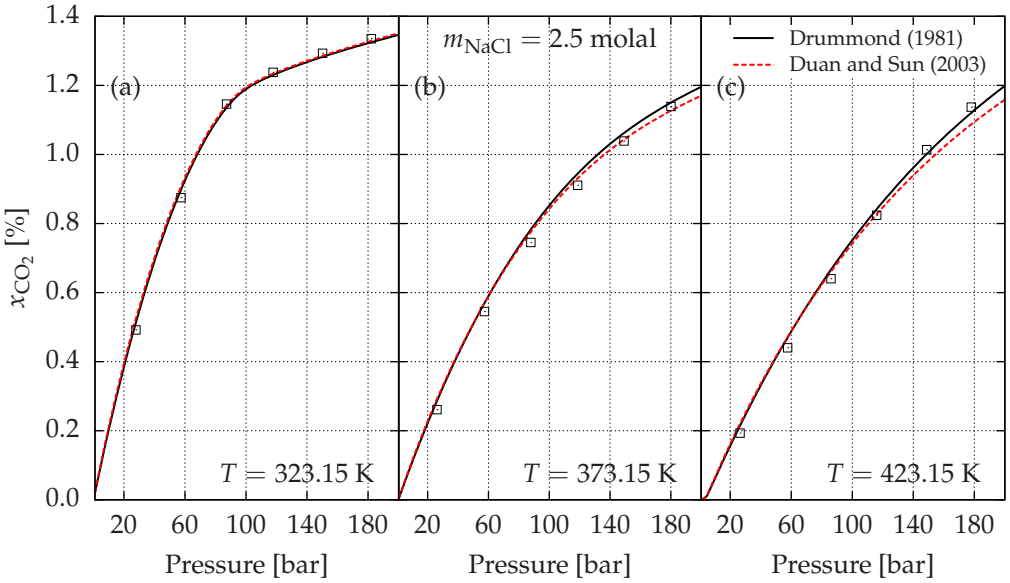


Figure 5.7: Comparison of our calculations of CO_2 solubility in 2.5 molal NaCl brine (lines), using the activity coefficient models of Duan and Sun (2003) and Drummond (1981) for CO_2(aq) and the fugacity coefficient model of Spycher et al. (2003) for CO_2(g) and H_2O(g), with the experimental solubility results of Hou et al. (2013b) (points). The calculations assumed a H_2O–CO_2–NaCl system composed of an aqueous and gaseous phase.

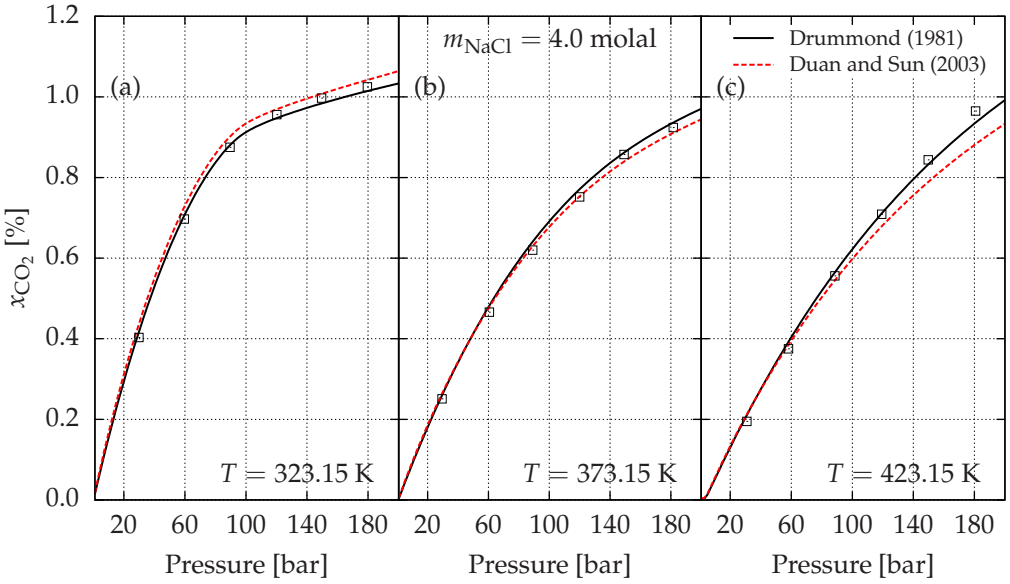


Figure 5.8: Comparison of our calculations of CO_2 solubility in 4.0 molal NaCl brine (lines), using the activity coefficient models of Duan and Sun (2003) and Drummond (1981) for CO_2(aq) and the fugacity coefficient model of Spycher et al. (2003) for CO_2(g) and H_2O(g), with the experimental solubility results of Hou et al. (2013b) (points). The calculations assumed a H_2O–CO_2–NaCl system composed of an aqueous and gaseous phase.

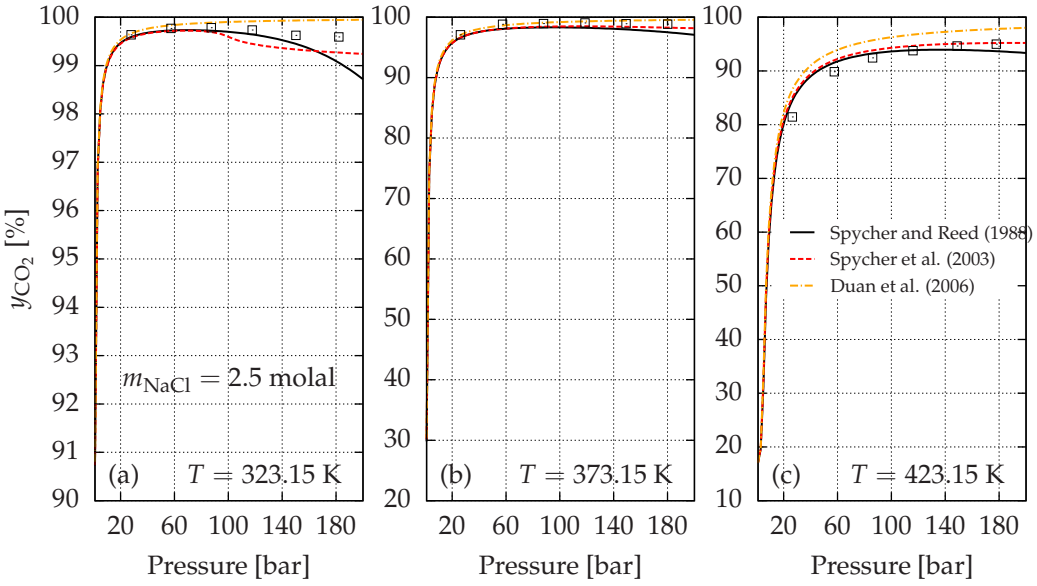


Figure 5.9: Comparison of our calculations of H_2O solubility (in terms of y_{CO_2}) in the CO_2-rich phase (lines), using the activity coefficient model of Duan and Sun (2003) for CO_2(aq) and the fugacity coefficient models of Spycher and Reed (1988), Spycher et al. (2003) and Duan et al. (2006) for CO_2(g) and H_2O(g), with the experimental solubility results of Hou et al. (2013b) (points). The calculations assumed a H_2O–CO_2–NaCl system composed of an aqueous and gaseous phase, where the aqueous phase contains 2.5 molal of NaCl.

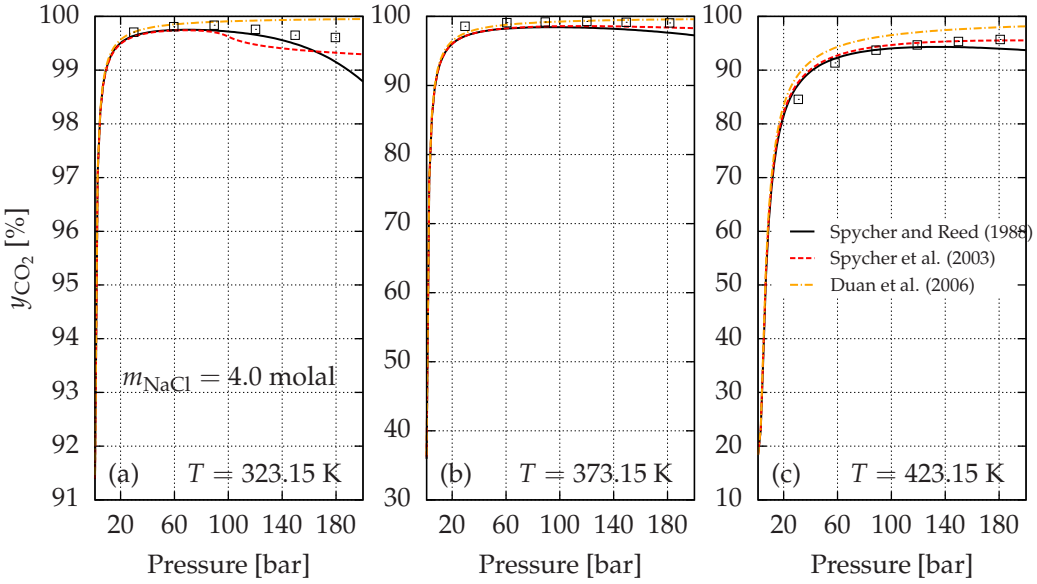


Figure 5.10: Comparison of our calculations of H_2O solubility (in terms of y_{CO_2}) in the CO_2-rich phase (lines), using the activity coefficient model of Duan and Sun (2003) for CO_2(aq) and the fugacity coefficient models of Spycher and Reed (1988), Spycher et al. (2003) and Duan et al. (2006) for CO_2(g) and H_2O(g), with the experimental solubility results of Hou et al. (2013b) (points). The calculations assumed a H_2O–CO_2–NaCl system composed of an aqueous and gaseous phase, where the aqueous phase contains 4.0 molal of NaCl.

Table 5.2: Comparison of our calculations of carbon dioxide solubility in pure water with the experimental solubility data of Hou et al. (2013a).

T (K)	P (bar)	x_{exp} (%)	x_{SR} (%)	x_{SP} (%)	x_{DSC} (%)	$Error_{SR}$ (%)	$Error_{SP}$ (%)	$Error_{DSC}$ (%)
349.19	252.2	2.148	2.0160	2.0244	2.0436	6.143	5.755	4.860
349.13	183.1	2.017	1.9125	1.8684	1.8783	5.183	7.368	6.878
374.41	72.1	1.105	1.0059	1.0296	1.0349	8.973	6.824	6.344
374.18	144.4	1.711	1.5580	1.5868	1.6012	8.940	7.260	6.418
374.91	188.6	1.924	1.7508	1.7782	1.7927	9.001	7.580	6.826
374.15	223.4	2.048	1.8449	1.8862	1.9126	9.919	7.900	6.613
374.99	272.6	2.189	1.9067	2.0123	2.0529	12.895	8.073	6.220
					Average	**8.722**	**7.251**	**6.308**

Note: In this and subsequent tables, x_{exp} denotes experimental molar fraction of aqueous CO_2. x_{SR}, x_{SP} and x_{DSC} denote the calculated molar fraction of aqueous CO_2 using respectively the fugacity coefficient models of Spycher and Reed (1988), Spycher et al. (2003) and Duan et al. (2006) for the gaseous mixture H_2O–CO_2. $Error_{SR}$, $Error_{SP}$ and $Error_{DSC}$ denote respectively the percentage deviation of x_{SR}, x_{SP} and x_{DSC} from x_{exp}.

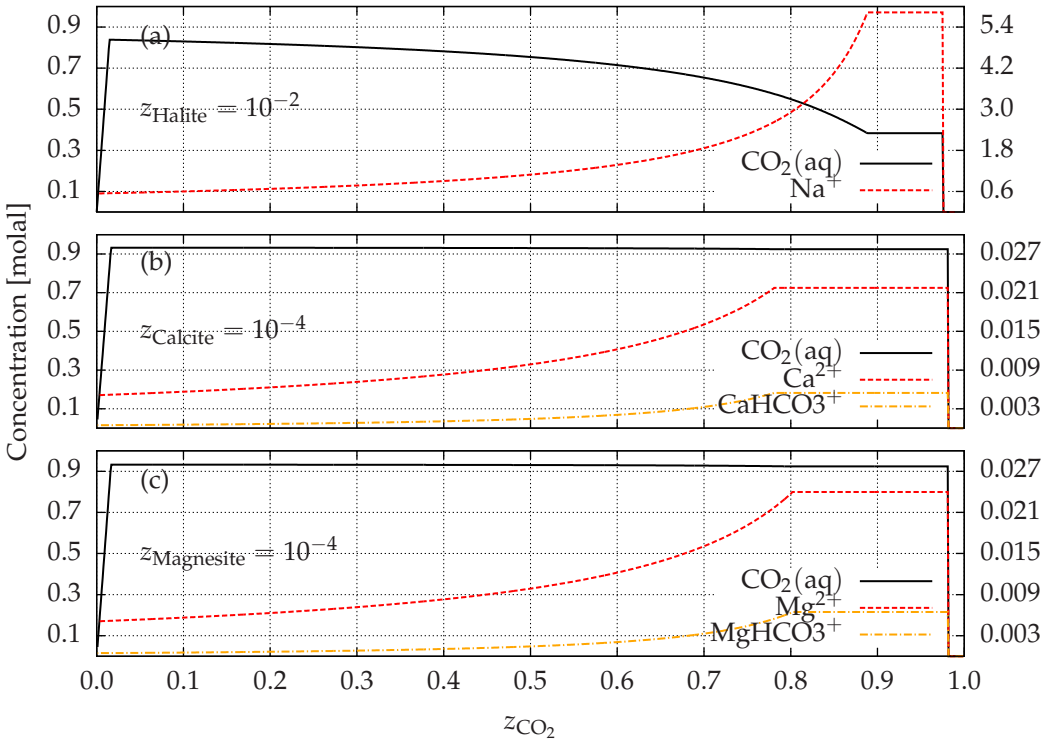


Figure 5.11: Phase behaviour of the systems H_2O–CO_2–Halite (a), H_2O–CO_2–Calcite (b) and H_2O–CO_2–Magnesite (c) respectively along the line of feed molar fraction $z_{Halite} = 10^{-2}$, $z_{Calcite} = 10^{-4}$ and $z_{Magnesite} = 10^{-4}$. Concentrations of CO_2(aq) are given on the left vertical axis and concentrations of other aqueous species on the right. The calculations assumed a H_2O–CO_2–Mineral system composed of an aqueous, gaseous and mineral phase.

Table 5.3: Comparison of our calculations of carbon dioxide solubility in NaCl brine, using the activity coefficient model of Duan and Sun (2003) for CO_2(aq), with the experimental solubility data of Hou et al. (2013b).

T (K)	P (bar)	m_{NaCl} (molal)	x_{exp} (%)	x_{SR} (%)	x_{SP} (%)	x_{DSC} (%)	$Error_{SR}$ (%)	$Error_{SP}$ (%)	$Error_{DSC}$ (%)
323.15	27.82	2.5	0.492	0.5046	0.5164	0.5141	2.567	4.951	4.488
	57.39		0.875	0.8666	0.8928	0.8848	0.965	2.034	1.117
	87.3		1.146	1.1115	1.1232	1.1160	3.008	1.986	2.618
	117.73		1.238	1.2761	1.2136	1.2090	3.077	1.970	2.345
	150.2		1.293	1.3866	1.2654	1.2530	7.242	2.136	3.093
	182.11		1.335	1.4460	1.3064	1.2920	8.312	2.145	3.220
						Average	**4.195**	**2.537**	**2.813**
373.15	26.13	2.5	0.254	0.2827	0.2862	0.2874	11.304	12.676	13.135
	57.42		0.532	0.5491	0.5610	0.5631	3.213	5.451	5.845
	87.89		0.739	0.7458	0.7641	0.7675	0.914	3.391	3.861
	118.67		0.899	0.8969	0.9167	0.9224	0.228	1.969	2.606
	149.21		1.032	1.0100	1.0268	1.0344	2.129	0.503	0.234
	180.13		1.139	1.0957	1.1099	1.1148	3.800	2.559	2.121
						Average	**3.598**	**4.425**	**4.634**
423.15	26.43	2.5	0.193	0.2360	0.2379	0.2401	22.272	23.259	24.430
	57.66		0.440	0.4560	0.4638	0.4744	3.626	5.404	7.810
	86.00		0.640	0.6364	0.6504	0.6672	0.569	1.622	4.249
	116.04		0.824	0.7939	0.8152	0.8397	3.653	1.068	1.901
	148.81		1.014	0.9331	0.9634	0.9972	7.980	4.994	1.652
	177.94		1.137	1.0321	1.0722	1.1149	9.230	5.700	1.944
						Average	**7.888**	**7.008**	**6.998**
323.15	29.83	4.0	0.403	0.4196	0.4298	0.4277	4.112	6.652	6.133
	59.54		0.689	0.6945	0.7153	0.7087	0.798	3.820	2.854
	89.53		0.871	0.8807	0.8873	0.8823	1.111	1.866	1.292
	120.17		0.956	1.0073	0.9537	0.9489	5.365	0.237	0.743
	149.59		0.997	1.0866	0.9919	0.9817	8.988	0.513	1.533
	179.54		1.025	1.1347	1.0246	1.0128	10.700	0.037	1.193
						Average	**5.179**	**2.188**	**2.291**
373.15	29.51	4.0	0.251	0.2562	0.2597	0.2606	2.076	3.471	3.836
	60.68		0.446	0.4618	0.4721	0.4737	3.553	5.860	6.204
	89.16		0.620	0.6058	0.6206	0.6231	2.288	0.103	0.503
	120.03		0.752	0.7263	0.7420	0.7461	3.420	1.331	0.780
	149.24		0.857	0.8135	0.8267	0.8321	5.079	3.535	2.910
	181.62		0.934	0.8869	0.8977	0.9009	5.048	3.889	3.544
						Average	**3.577**	**3.031**	**2.963**
423.15	30.93	4.0	0.195	0.2219	0.2239	0.2262	13.786	14.830	16.020
	58.16		0.375	0.3732	0.3795	0.3875	0.482	1.190	3.346
	88.57		0.556	0.5261	0.5377	0.5508	5.376	3.299	0.928
	119.22		0.709	0.6524	0.6697	0.6890	7.990	5.536	2.827
	149.79		0.844	0.7554	0.7793	0.8053	10.503	7.664	4.586
	180.79		0.972	0.8397	0.8720	0.9051	13.611	10.293	6.879
						Average	**8.625**	**7.135**	**5.764**

Note: m_{NaCl} denotes the salinity of the NaCl brine in molality scale.

Table 5.4: Comparison of our calculations of carbon dioxide solubility in NaCl brine, using the activity coefficient model of Drummond (1981) for CO_2(aq), with the experimental solubility data of Hou et al. (2013b).

T (K)	P (bar)	m_{NaCl} (molal)	x_{exp} (%)	x_{SR} (%)	x_{SP} (%)	x_{DSC} (%)	$Error_{SR}$ (%)	$Error_{SP}$ (%)	$Error_{DSC}$ (%)
323.15	27.82	2.5	0.492	0.4938	0.5053	0.5030	0.362	2.695	2.242
	57.39		0.875	0.8580	0.8840	0.8761	1.941	1.029	0.120
	87.30		1.146	1.1059	1.1176	1.1104	3.496	2.479	3.108
	117.73		1.238	1.2723	1.2100	1.2054	2.770	2.261	2.635
	150.20		1.293	1.3832	1.2623	1.2499	6.976	2.378	3.332
	182.11		1.335	1.4416	1.3024	1.2881	7.983	2.441	3.514
						Average	**3.921**	**2.214**	**2.492**
373.15	26.13	2.5	0.254	0.2777	0.2811	0.2822	9.318	10.665	11.115
	57.42		0.532	0.5492	0.5611	0.5632	3.232	5.470	5.864
	87.89		0.739	0.7524	0.7709	0.7744	1.819	4.318	4.792
	118.67		0.899	0.9099	0.9300	0.9358	1.214	3.443	4.089
	149.21		1.032	1.0283	1.0454	1.0531	0.358	1.297	2.048
	180.13		1.139	1.1183	1.1328	1.1378	1.815	0.548	0.102
						Average	**2.959**	**4.290**	**4.668**
423.15	26.43	2.5	0.193	0.2296	0.2314	0.2336	18.956	19.916	21.055
	57.66		0.440	0.4541	0.4619	0.4724	3.198	4.968	7.364
	86.00		0.640	0.6414	0.6555	0.6724	0.213	2.422	5.069
	116.04		0.824	0.8074	0.8290	0.8539	2.019	0.610	3.629
	148.81		1.014	0.9560	0.9870	1.0218	5.717	2.658	0.765
	177.94		1.137	1.0632	1.1045	1.1485	6.495	2.860	1.007
						Average	**6.100**	**5.572**	**6.482**
323.15	29.83	4.0	0.403	0.4010	0.4108	0.4088	0.505	1.924	1.427
	59.54		0.689	0.6743	0.6946	0.6881	2.128	0.807	0.131
	89.53		0.871	0.8598	0.8663	0.8614	1.282	0.544	1.104
	120.17		0.956	0.9846	0.9323	0.9275	2.995	2.482	2.976
	149.59		0.997	1.0609	0.9684	0.9585	6.413	2.864	3.861
	179.54		1.025	1.1050	0.9978	0.9862	7.801	2.657	3.783
						Average	**3.521**	**1.880**	**2.214**
373.15	29.51	4.0	0.251	0.2521	0.2556	0.2565	0.449	1.822	2.181
	60.68		0.446	0.4653	0.4757	0.4773	4.338	6.661	7.008
	89.16		0.620	0.6166	0.6317	0.6342	0.553	1.880	2.287
	120.03		0.752	0.7435	0.7596	0.7638	1.128	1.011	1.574
	149.24		0.857	0.8352	0.8488	0.8543	2.545	0.961	0.318
	181.62		0.934	0.9117	0.9229	0.9262	2.384	1.193	0.838
						Average	**1.900**	**2.254**	**2.368**
423.15	30.93	4.0	0.195	0.2201	0.2221	0.2244	12.870	13.906	15.087
	58.16		0.375	0.3803	0.3867	0.3949	1.411	3.116	5.313
	88.57		0.556	0.5450	0.5570	0.5706	1.973	0.178	2.634
	119.22		0.709	0.6830	0.7012	0.7213	3.669	1.101	1.734
	149.79		0.844	0.7966	0.8219	0.8492	5.615	2.623	0.622
	180.79		0.972	0.8903	0.9245	0.9596	8.407	4.890	1.273
						Average	**5.658**	**4.302**	**4.444**

Note: m_{NaCl} denotes the salinity of the NaCl brine in molality scale.

Table 5.5: Comparison of our calculations of carbon dioxide solubility in NaCl+KCl brine, using the fugacity coefficient model of Duan et al. (2006) for CO_2(g), with the experimental solubility data of Tong et al. (2013).

T (K)	P (bar)	x_{exp} (%)	x_{DS} (%)	x_{Dr} (%)	$Error_{DS}$ (%)	$Error_{Dr}$ (%)
309.14	11.9	0.384	0.4212	0.4129	9.700	7.532
308.90	37.4	1.180	1.1193	1.1070	5.148	6.187
324.11	10.7	0.322	0.2789	0.2724	13.384	15.419
324.10	43.5	1.062	0.9387	0.9275	11.615	12.663
343.83	17.7	0.295	0.3373	0.3293	14.334	11.624
343.92	36.1	0.642	0.6314	0.6205	1.656	3.349
343.88	136.9	1.580	1.4427	1.4324	8.688	9.342
345.04	102.5	1.381	1.2803	1.2687	7.290	8.135
374.92	29.9	0.409	0.4063	0.3950	0.668	3.421
374.92	69.6	0.872	0.8259	0.8103	5.290	7.078
374.89	147.8	1.415	1.3196	1.3046	6.744	7.802
424.67	40.0	0.407	0.4382	0.4197	7.658	3.113
424.62	86.8	0.869	0.8625	0.8356	0.752	3.849
424.64	171.6	1.460	1.4174	1.3890	2.920	4.863
				Average	**6.846**	**7.455**

Note: The salinity of the NaCl+KCl brine is $m_{NaCl} = 0.910$ molal and $m_{KCl} = 0.143$ molal. x_{DS} and x_{Dr} denote the calculated aqueous molar fraction of CO_2 using respectively the activity coefficient models of Duan and Sun (2003) and Drummond (1981) for CO_2(aq). $Error_{DS}$ and $Error_{Dr}$ denote respectively the percentage deviation of x_{DS} and x_{Dr} from x_{exp}.

Table 5.6: Comparison of our calculations of carbon dioxide solubility in $MgCl_2$ brine, using the fugacity coefficient model of Duan et al. (2006) for CO_2(g), with the experimental solubility data of Tong et al. (2013).

T (K)	P (bar)	m_{MgCl_2} (molal)	x_{exp} (%)	x_{DS} (%)	x_{Dr} (%)	$Error_{DS}$ (%)	$Error_{Dr}$ (%)
309.58	12.5	1.0	0.382	0.3629	0.2850	5.007	25.387
309.83	42.4	1.0	1.085	0.9930	0.7949	8.482	26.740
324.41	16.5	1.0	0.366	0.3470	0.2805	5.193	23.351
324.37	56.8	1.0	1.041	0.9365	0.7719	10.036	25.850
343.90	20.4	1.0	0.344	0.3234	0.2668	6.002	22.430
344.20	76.2	1.0	0.996	0.9118	0.7696	8.456	22.727
344.93	305.8	1.0	1.532	1.4529	1.2439	5.162	18.806
374.24	26.4	1.0	0.294	0.3076	0.2578	4.612	12.308
374.22	102.5	1.0	1.005	0.9023	0.7776	10.219	22.630
374.91	349.3	1.0	1.609	1.4844	1.3090	7.744	18.644
424.03	39.5	1.0	0.345	0.3632	0.3122	5.265	9.495
423.95	126.3	1.0	0.980	0.9498	0.8448	3.083	13.800
424.63	197.4	1.0	1.310	1.2619	1.1400	3.673	12.977
424.63	283.7	1.0	1.609	1.5032	1.3778	6.578	14.367
344.68	130.9	5.0	0.370	0.3372	0.3512	8.871	5.092
344.98	312.0	5.0	0.471	0.4477	0.4406	4.941	6.457
374.72	75.5	5.0	0.257	0.2259	0.2556	12.112	0.558
374.68	156.0	5.0	0.384	0.3425	0.3962	10.804	3.174
375.02	205.8	5.0	0.414	0.3851	0.4441	6.972	7.276
424.39	47.0	5.0	0.131	0.1327	0.1607	1.301	22.678
424.33	103.4	5.0	0.231	0.2465	0.3191	6.721	38.138
424.49	161.8	5.0	0.355	0.3330	0.4421	6.194	24.539
					Average	**6.701**	**17.156**

Note: m_{MgCl_2} denotes the salinity of the $MgCl_2$ brine in molality scale.

Table 5.7: Comparison of our calculations of carbon dioxide solubility in $CaCl_2$ brine, using the fugacity coefficient model of Duan et al. (2006) for CO_2(g), with the experimental solubility data of Tong et al. (2013).

T (K)	P (bar)	m_{CaCl_2} (molal)	x_{exp} (%)	x_{DS} (%)	x_{Dr} (%)	$Error_{DS}$ (%)	$Error_{Dr}$ (%)
309.67	15.3	1.0	0.448	0.4343	0.3490	3.058	22.091
309.28	40.1	1.0	1.036	0.9639	0.7854	6.962	24.191
344.67	25.8	1.0	0.402	0.3940	0.3319	1.983	17.449
344.35	73.0	1.0	0.976	0.8864	0.7596	9.185	22.177
344.95	310.2	1.0	1.510	1.4567	1.2668	3.527	16.104
374.70	31.3	1.0	0.377	0.3580	0.3068	5.034	18.619
374.70	100.2	1.0	0.934	0.8887	0.7798	4.853	16.510
374.88	373.8	1.0	1.613	1.5136	1.3603	6.163	15.669
424.13	43.9	1.0	0.392	0.3968	0.3441	1.213	12.225
424.10	127.7	1.0	1.001	0.9573	0.8567	4.362	14.417
424.43	268.2	1.0	1.581	1.4639	1.3462	7.409	14.854
424.40	82.9	3.0	0.380	0.3508	0.2967	7.687	21.931
424.39	271.1	3.0	0.807	0.7236	0.6605	10.332	18.149
424.38	379.9	3.0	0.903	0.8315	0.7737	7.919	14.318
344.72	61.2	5.0	0.272	0.2257	0.2308	17.023	15.131
344.70	145.4	5.0	0.357	0.3480	0.3621	2.519	1.428
344.96	345.2	5.0	0.511	0.4615	0.4467	9.692	12.583
374.72	76.1	5.0	0.266	0.2251	0.2550	15.373	4.118
374.72	169.1	5.0	0.402	0.3508	0.4061	12.741	1.008
374.72	342.5	5.0	0.484	0.4724	0.5242	2.403	8.313
424.43	46.4	5.0	0.141	0.1311	0.1586	7.019	12.505
424.42	105.3	5.0	0.239	0.2493	0.3233	4.323	35.271
					Average	**6.854**	**15.412**

Note: m_{CaCl_2} denotes the salinity of the $CaCl_2$ brine in molality scale.

5.3 Comparison with Other Geochemical Solvers

This section compares the accuracy of the calculations in this work with other geochemical solvers. The comparison is performed for two quantities: the solubility of carbon dioxide in NaCl brines, and the pH in CO_2 saturated NaCl brines. The objective is to verify how well these solvers reproduce the recent solubility experiments of Hou et al. (2013b) and pH measurements of Peng et al. (2013) at high temperatures, pressures and salinities. Thus, this analysis should determine under which conditions the assessed solvers provide reliable estimates.

The geochemical package PHREEQC (Parkhurst and Appelo, 2013) was used for both solubility and pH comparisons. The well-known solubility model of Duan and Sun (2003) was used only for the solubility comparison. In a recent update of PHREEQC, several new modelling features were implemented, where Pitzer aqueous models are used for high-salinity brines and the Peng-Robinson equation of state is used for calculating the fugacity coefficients of gases at high pressures, allowing for a more accurate gas solubility calculation. The thermodynamic model developed by Duan and Sun (2003), based on the specific interaction model of Pitzer with a parametrisation of a large amount of experimental data in the literature, allows the calculation of CO_2 solubility in NaCl brines for temperatures 273–533 K, pressures 0–2000 bar and salinities 0–4.3 molal.

Table 5.8 presents a comparison of the calculated solubility results with the experimental data of Hou et al. (2013b). The results produced by this work used the activity coefficient model of

Drummond (1981) for CO_2(aq) and the fugacity coefficient model of Spycher et al. (2003) for CO_2(g) and H_2O(g).

It can be observed from Table 5.8 that PHREEQC (Parkhurst and Appelo, 2013) fails to estimate the solubility of CO_2 at higher salinities accurately. For 4.0 molal NaCl brines, solubility results obtained using PHREEQC have percentage deviations of 18% in average, taking all calculations at that salinity. At extreme conditions of temperature, pressure and salinity, a percentage deviation error of 26% was achieved with the same solver.

The solubility model of Duan and Sun (2003) performed better, even at high temperatures, pressures and salinities. However, the use of a proper selection of activity and fugacity coefficient models in this work yielded the most accurate solubility results for most of the temperature, pressure and salinity conditions, as seen in Table 5.8.

Table 5.9 presents a comparison of the calculated pH in CO_2 saturated NaCl brines using our solver with PHREEQC (Parkhurst and Appelo, 2013). The calculations are compared against the experimental measurements of Peng et al. (2013). So far, all calculations in this work have adopted the HKF (Helgeson et al., 1981) activity coefficient model for solvent water and ionic species. Although it produces accurate carbon dioxide solubility results even at high saline NaCl brines, this model turns out not to be sufficiently accurate for pH calculations. Therefore, the Pitzer's based HMW activity coefficient model (Harvie et al., 1984) was used for the pH calculations, which is the same model adopted for the calculations using PHREEQC.

The only difference on the calculations in Table 5.9 is the adopted thermodynamic database. Here the SUPCRT (Johnson et al., 1992) database is used, while PHREEQC uses its own database. Thus, we see that the use of the SUPCRT database results in more accurate pH calculations than when the PHREEQC database is used. The only exception is at low temperatures and high pressures, where the use of the SUPCRT database can result in errors in the order of 9%.

5.4 Sequential Chemical Equilibrium Calculations

Efficient sequential equilibrium calculations are important for many critical applications, such as reactive transport modelling. This section shows the performance results of the Gibbs energy minimisation algorithm presented in Chapter 3. It also demonstrates the importance of the watchdog strategy and scaling of the variables on the efficiency of the equilibrium calculations.

Consider the multiphase system in Table 5.10. This system is of interest for modelling water-gas-rock interactions in CO_2 storage in deep saline aquifers, where the reservoir is mainly composed of carbonate minerals. The formation rock is modelled by the mineral phases calcite, magnesite and dolomite. To model a possible precipitation of sodium chloride when the aqueous phase becomes salt saturated, the mineral phase halite is also included in the formulation.

To assess the efficiency of the equilibrium calculations in a carbon injection simulation, a chemical state initially containing only an aqueous phase and mineral phases is considered. Then, CO_2 is gradually added to the system until a specified amount is reached, which should be high enough to saturate the aqueous phase and produce a gaseous phase. This example is also useful to assess the robustness of the algorithm when handling phase assemblage transitions.

Table 5.8: Comparison of our calculations of carbon dioxide solubility in NaCl brine with the calculations using PHREEQC (Parkhurst and Appelo, 2013) and the solubility model of Duan and Sun (2003) as well as with the experimental solubility data of Hou et al. (2013b).

T (K)	P (bar)	m_{NaCl} (molal)	m_{exp} (molal)	m_{PHC} (molal)	m_{Duan} (molal)	m_{this} (molal)	$\text{Error}_{\text{PHC}}$ (%)	$\text{Error}_{\text{Duan}}$ (%)	$\text{Error}_{\text{this}}$ (%)
323.15	27.82	2.5	0.2744	0.2716	0.3016	0.2819	1.038	9.893	2.709
	57.39		0.4900	0.4738	0.5233	0.4951	3.303	6.800	1.038
	87.3		0.6435	0.5986	0.6645	0.6274	6.977	3.264	2.507
	117.73		0.6958	0.6581	0.7229	0.6799	5.419	3.894	2.289
	150.2		0.7271	0.6948	0.7518	0.7096	4.445	3.394	2.409
	182.11		0.7511	0.7237	0.7788	0.7325	3.643	3.694	2.473
						Average	**4.137**	**5.156**	**2.237**
373.15	26.13	2.5	0.1413	0.1424	0.1682	0.1565	0.743	18.996	10.695
	57.42		0.2969	0.2860	0.3324	0.3132	3.665	11.964	5.501
	87.89		0.4133	0.3952	0.4558	0.4312	4.370	10.294	4.352
	118.67		0.5035	0.4814	0.5504	0.5210	4.398	9.305	3.476
	149.21		0.5788	0.5475	0.6198	0.5864	5.410	7.081	1.311
	180.13		0.6395	0.6005	0.6696	0.6360	6.101	4.703	0.555
						Average	**4.115**	**10.391**	**4.315**
423.15	26.43	2.5	0.1073	0.1012	0.1241	0.1288	5.718	15.617	19.962
	57.66		0.2453	0.2267	0.2749	0.2576	7.588	12.060	4.991
	86.00		0.3575	0.3227	0.3904	0.3663	9.744	9.191	2.438
	116.04		0.4612	0.4135	0.4947	0.4640	10.340	7.267	0.615
	148.81		0.5686	0.4991	0.5909	0.5534	12.226	3.919	2.685
	177.94		0.6384	0.5652	0.6635	0.6199	11.464	3.934	2.892
						Average	**9.513**	**8.665**	**5.597**
323.15	29.83	4.0	0.2246	0.2023	0.2506	0.2289	9.930	11.575	1.932
	59.54		0.3851	0.3413	0.4183	0.3882	11.374	8.620	0.812
	89.53		0.4877	0.4266	0.5240	0.4850	12.532	7.438	0.549
	120.17		0.5358	0.4674	0.5658	0.5224	12.762	5.603	2.505
	149.59		0.5590	0.4908	0.5872	0.5428	12.198	5.047	2.892
	179.54		0.5748	0.5101	0.6084	0.5594	11.264	5.836	2.684
						Average	**11.677**	**7.353**	**1.896**
373.15	29.51	4.0	0.1397	0.1121	0.1523	0.1422	19.743	9.038	1.827
	60.68		0.2487	0.2101	0.2792	0.2653	15.512	12.275	6.693
	89.16		0.3463	0.2803	0.3692	0.3528	19.058	6.614	1.891
	120.03		0.4206	0.3407	0.4442	0.4249	18.993	5.615	1.019
	149.24		0.4798	0.3853	0.4973	0.4752	19.698	3.644	0.969
	181.62		0.5233	0.4243	0.5397	0.5170	18.923	3.128	1.204
						Average	**18.655**	**6.719**	**2.267**
423.15	30.93	4.0	0.1085	0.0860	0.1199	0.1236	21.587	10.556	13.937
	58.16		0.2089	0.1606	0.2232	0.2155	23.135	6.826	3.128
	88.57		0.3104	0.2313	0.3210	0.3109	25.471	3.432	0.179
	119.22		0.3964	0.2955	0.4044	0.3920	25.447	2.028	1.109
	149.79		0.4725	0.3510	0.4753	0.4600	25.710	0.598	2.645
	180.79		0.5448	0.4035	0.5367	0.5179	26.583	1.493	4.936
						Average	**24.656**	**4.155**	**4.322**

Note: m_{NaCl} denotes the salinity of the NaCl brine in molality scale. m_{PHC} and m_{Duan} denote the calculated molality of CO_2 using PHREEQC (Parkhurst and Appelo, 2013) and the solubility model of Duan and Sun (2003) respectively. m_{this} denotes the calculated molality of CO_2 using our solver with the activity coefficient model of Drummond (1981) for CO_2(aq) and the fugacity coefficient model of Spycher et al. (2003) for CO_2(g) and H_2O(g). $\text{Error}_{\text{PHC}}$, $\text{Error}_{\text{Duan}}$ and $\text{Error}_{\text{this}}$ denote respectively the percentage deviation of m_{PHC}, m_{Duan} and m_{this} from m_{exp}.

Table 5.9: Comparison of our calculations of pH in CO_2 saturated NaCl brine (1 molal) with the calculations using PHREEQC (Parkhurst and Appelo, 2013) as well as with the experimental pH measurements of Peng et al. (2013).

T (K)	P (MPa)	pH_{exp}	pH_{PHC}	pH_{this}	$Error_{PHC}$ (%)	$Error_{this}$ (%)
308.2	0.37	3.58	3.567	3.574	0.36	0.17
308.3	0.60	3.52	3.459	3.472	1.73	1.36
308.0	0.99	3.35	3.353	3.367	0.09	0.51
308.3	2.44	3.06	3.160	3.190	3.27	4.26
308.2	6.15	2.81	2.960	3.039	5.34	8.15
308.3	9.27	2.76	2.870	3.003	3.99	8.81
308.3	15.38	2.72	2.770	2.975	1.84	9.39
				Average	**2.37**	**4.67**
323.0	0.37	3.73	3.630	3.631	2.68	2.65
323.0	0.61	3.60	3.524	3.525	2.11	2.08
323.2	0.99	3.46	3.420	3.425	1.16	1.02
323.3	2.43	3.21	3.224	3.246	0.44	1.11
323.3	6.23	3.01	3.023	3.083	0.43	2.44
323.3	9.26	2.97	2.940	3.034	1.01	2.15
323.0	15.36	2.92	2.830	2.994	3.08	2.54
				Average	**1.56**	**2.00**
343.2	0.37	3.83	3.730	3.719	2.61	2.90
343.0	0.61	3.68	3.620	3.612	1.63	1.85
343.0	1.00	3.55	3.512	3.508	1.07	1.19
343.0	2.43	3.32	3.317	3.328	0.09	0.23
343.1	6.23	3.15	3.115	3.157	1.11	0.23
343.1	9.26	3.10	3.030	3.099	2.26	0.05
343.0	15.39	3.04	2.930	3.042	3.62	0.06
				Average	**1.77**	**0.93**
363.0	0.60	3.77	3.721	3.705	1.3	1.73
363.0	1.00	3.64	3.613	3.597	0.74	1.19
363.0	2.44	3.39	3.420	3.413	0.88	0.69
363.0	6.24	3.22	3.212	3.237	0.25	0.52
363.0	9.28	3.17	3.130	3.172	1.26	0.07
363.0	15.30	3.12	3.020	3.104	3.21	0.50
				Average	**1.27**	**0.78**
373.0	0.60	3.80	3.780	3.748	0.53	1.36
373.0	1.00	3.66	3.667	3.640	0.19	0.55
372.9	2.44	3.42	3.470	3.455	1.46	1.03
373.0	6.25	3.26	3.263	3.276	0.09	0.50
373.2	9.28	3.20	3.180	3.210	0.63	0.33
373.2	15.30	3.13	3.070	3.138	1.92	0.25
				Average	**0.80**	**0.67**

Note: pH_{PHC} denotes the calculated pH using PHREEQC (Parkhurst and Appelo, 2013). pH_{this} denotes the calculated pH using this work's solver with the Pitzer activity coefficient model for solvent water, ionic species and the neutral species CO_2(aq), and the fugacity coefficient model of Spycher et al. (2003) for CO_2(g) and H_2O(g). $Error_{PHC}$ and $Error_{this}$ denote respectively the percentage deviation of pH_{PHC} and pH_{this} from pH_{exp}.

The modelling of the previous problem is now described. Let H_2O, CO_2, NaCl, $CaCO_3$ and $MgCO_3$ denote auxiliary components of the chemical system in Table 5.10. Consider their molar abundance at the initial and final states as given in Table 5.11. Next, let $\mathbf{n}_c$ denote the vector of molar abundance of the components. Define the following linear path:

$$\mathbf{n}_c(t) := \mathbf{n}_c^{\mathrm{i}} + t(\mathbf{n}_c^{\mathrm{f}} - \mathbf{n}_c^{\mathrm{i}}), \qquad (5.4)$$

where $\mathbf{n}_c^{\mathrm{i}}$ and $\mathbf{n}_c^{\mathrm{f}}$ are the given initial and final molar abundance of the components; and $t \in [0, 1]$ is a scalar parameter. The inputs of our sequential equilibrium calculations are determined, therefore, by gradually increasing the parameter t in order to model the addition or removal of components from the system. Note, however, that the molar abundance of the components $\mathbf{n}_c$ are auxiliary inputs for the chemical equilibrium calculations. The natural inputs are the number of moles of the chemical elements, which can be determined from the expressions in Table 5.12.

Using the molar abundance of the components in Table 5.11, the initial and final equilibrium states of the system were calculated as shown in Table 5.13. It can be seen that the amount of dissolved carbon increased from the initial to the final state, while the aqueous solution became more acidic, with initial and final pH as 9.2 and 4.8 respectively. In addition, note that the zeroed number of moles of some species indicates that the phases containing them are not present at equilibrium. For example, from all assumed mineral phases, only calcite and dolomite exist at given conditions of temperature, pressure and molar amounts of the components.

Table 5.10: Description of the chemical system H_2O–CO_2–NaCl–$CaCO_3$–$MgCO_3$ with their phases and respective chemical species.

Aqueous Phase	**Gaseous Phase**
$H_2O(l)$	$H_2O(g)$
H^+	$CO_2(g)$
OH^-	**Mineral Phase #1**
HCO_3^-	$NaCl(s)$ (*Halite*)
CO_3^{2-}	**Mineral Phase #2**
Na^+	$CaCO_3(s)$ (*Calcite*)
Cl^-	**Mineral Phase #3**
Ca^{2+}	$MgCO_3(s)$ (*Magnesite*)
Mg^{2+}	**Mineral Phase #4**
$MgCl^+$	$CaMg(CO_3)_2(s)$ (*Dolomite*)
$CaCl^+$	
$CO_2(aq)$	
$CaCO_3(aq)$	
$MgCO_3(aq)$	
$CaCl_2(aq)$	

Table 5.11: The number of moles of each component at both initial and final states.

Components	Initial state [mol]	Final state [mol]
H_2O	55.508	55.508
CO_2	0	2
NaCl	2	2
$CaCO_3$	5	5
$MgCO_3$	1	1

Table 5.12: Expressions for the molar amounts of the chemical elements from the molar amounts of the components.

Element	Expression
H	$2n_{H_2O}$
O	$n_{H_2O} + 2n_{CO_2} + 3n_{CaCO_3} + 3n_{MgCO_3}$
C	$n_{CO_2} + n_{CaCO_3} + n_{MgCO_3}$
Na	n_{NaCl}
Cl	n_{NaCl}
Ca	n_{CaCO_3}
Mg	n_{MgCO_3}

The calculations assumed $T = 60\,^{\circ}\mathrm{C}$ and $P = 150\,\mathrm{bar}$, and a tolerance error of 10^{-8}. In addition, the activity coefficient model of Duan and Sun (2003) was used for the aqueous species $CO_2(aq)$ because of the presence of ions Ca^{2+} and Mg^{2+} in the aqueous solution. The fugacity coefficient model of Duan et al. (2006) was adopted for the gaseous species $CO_2(g)$.

Table 5.13: Initial and final equilibrium states of the chemical system H_2O–CO_2–NaCl–$CaCO_3$–$MgCO_3$ at $T = 60\,°C$ and $P = 150\,bar$.

Species	Initial state [mol]	Final state [mol]
Aqueous Phase		
$H_2O(l)$	55.507721	55.462962
H^+	$7.051770 \cdot 10^{-10}$	$1.716270 \cdot 10^{-5}$
OH^-	$2.791921 \cdot 10^{-4}$	$1.122305 \cdot 10^{-8}$
HCO_3^-	$2.789948 \cdot 10^{-4}$	0.087354
$CO_2(aq)$	$9.827989 \cdot 10^{-8}$	0.741635
Na^+	1.765120	1.767785
Cl^-	1.765011	1.759351
$NaCl(aq)$	0.234880	0.232215
Ca^{2+}	$4.382456 \cdot 10^{-4}$	0.034912
Mg^{2+}	$1.783294 \cdot 10^{-5}$	0.001406
CO_3^{2-}	$2.714616 \cdot 10^{-4}$	$3.501025 \cdot 10^{-6}$
$CaCl^+$	$7.716976 \cdot 10^{-5}$	0.006016
$MgCl^+$	$3.341396 \cdot 10^{-6}$	$2.571146 \cdot 10^{-4}$
$CaCO_3(aq)$	$7.012652 \cdot 10^{-6}$	$6.919652 \cdot 10^{-6}$
$MgCO_3(aq)$	$1.049437 \cdot 10^{-7}$	$1.035521 \cdot 10^{-7}$
$CaCl_2(aq)$	$1.396495 \cdot 10^{-5}$	0.001081
Gaseous Phase		
$H_2O(g)$	0.000000	0.001352
$CO_2(g)$	0.000000	1.214680
Mineral Phase #1		
Halite	0.000000	0.000000
Mineral Phase #2		
Calcite	3.999485	3.959648
Mineral Phase #3		
Magnesite	0.000000	0.000000
Mineral Phase #4		
Dolomite	0.999979	0.998337

Figures 5.12 and 5.13 show the efficiency of the Gibbs energy minimisation algorithm *without* and *with* the watchdog strategy. They show the required number of iterations to achieve convergence for each of the 20 sequential equilibrium calculations. Observe that the watchdog strategy is capable of boosting the convergence speed significantly. For example, the first calculation required 3.5 times fewer iterations when adopting this strategy.

Note that the first calculation requires more iterations than the others. This is because only poor initial guesses for the molar abundance of the species and the Lagrange multipliers are available. The second calculation still requires a few more iterations than the others, which is justified by the fact that the initial state is relatively distant from the second, since the former did not contain any CO_2. Nevertheless, observe that the calculations achieve convergence in 1–3 iterations after the second calculation.

At about $n_{CO_2} = 0.8$ mol, the gaseous phase emerges in the system and the number of iterations increases slightly during the handling of this phase assemblage transition, as seen in Figure 5.12. As shown in Figure 5.13, however, the number of iterations in this region is just slightly affected when the watchdog strategy is used. This is an important efficiency demonstration of the method, since in multiphase reactive transport simulations the front of the flow is constantly experiencing appearance and disappearance of phases.

Figures 5.12 and 5.13 adopted the scaling procedure presented in Section 3.3.5. Figures 5.14 and 5.15, on the other hand, show the same calculations without such scaling. Figure 5.14 indicates that the trust-region minimisation algorithm alone is highly sensitive to scaling, as discussed in Nocedal and Wright (1999), and its efficiency is severely compromised without it. The watchdog strategy, however, is just slightly affected by the lack of scaling, with its main inefficiency occurring near a phase boundary as shown in Figure 5.15. Therefore, we see that scaling of the variables is important for efficient equilibrium calculations, either using or not the watchdog strategy, since any small gain in efficiency can cause expressive performance results in reactive transport simulations.

Figures 5.16 and 5.17 show the efficiency of the method when performing 100 sequential calculations instead of 20. Since these calculations used shorter subintervals, it can be said that the difference between two of its consecutive solutions is smaller than the one when 20 sequential calculations are performed. This is reflected by the decrease in the number of iterations in the region $[0.2, 0.8]$ when compared to the same region in Figures 5.12 and 5.13. Finally, increasing the number of subintervals to 1000 yields an average number of iterations of 1.2 with the use of the watchdog strategy.

Figure 5.18 demonstrate that the sequence of equilibrium calculations are still efficiently calculated assuming simultaneous variations in temperature and pressure. The initial and final temperatures and pressures used were 60–160 °C and 100–300 bar, resulting in a variation of 5 °C and 10 bar for every equilibrium calculation. Note that the number of iterations after the gaseous phase is formed has increased by one. This is because this phase is more sensitive to changes in temperature and pressure, and an extra iteration is necessary to correct the molar abundance of the gaseous species.

These results demonstrate that the use of the watchdog technique is vital if this algorithm is to be used for multiphase reactive transport modelling. By adopting this non-monotone strategy, the calculations will potentially converge in only few iterations, even in regions near the front flow. Moreover, the scaling technique used in the calculations is capable of further decreasing the number of iterations necessary to solve the chemical equilibrium problems. As to the accuracy of the calculations, the mass-balance residuals are either zero or in the range 10^{-25} to 10^{-19} for some elements. Thus, the accuracy of the method satisfies the relative threshold of 10^{-13} discussed in Kulik et al. (2013), which is necessary for avoiding accumulation of errors in a reactive transport simulation over time.

In Figures 5.12–5.18, the additional effort in correcting the mass-balance of the stable phases once an equilibrium calculation has finished was not represented. In these sequential calculations, only a single correction iteration was performed when unstable phases were encountered.

Finally, Figure 5.19 shows the pH of the aqueous solution and the molalities of the species Ca^{2+} and Mg^{2+} as carbon dioxide is injected into the system. Observe that the aqueous phase becomes acidic with the injection of CO_2, and promotes further dissolution of the carbonate minerals, as suggested by the increase in the concentrations of the ions Ca^{2+} and Mg^{2+}. Also note that once the gaseous phase emerges in the system, the solubility of CO_2 in the aqueous phase remains constant, and so does the pH and the concentrations of the ionic species.

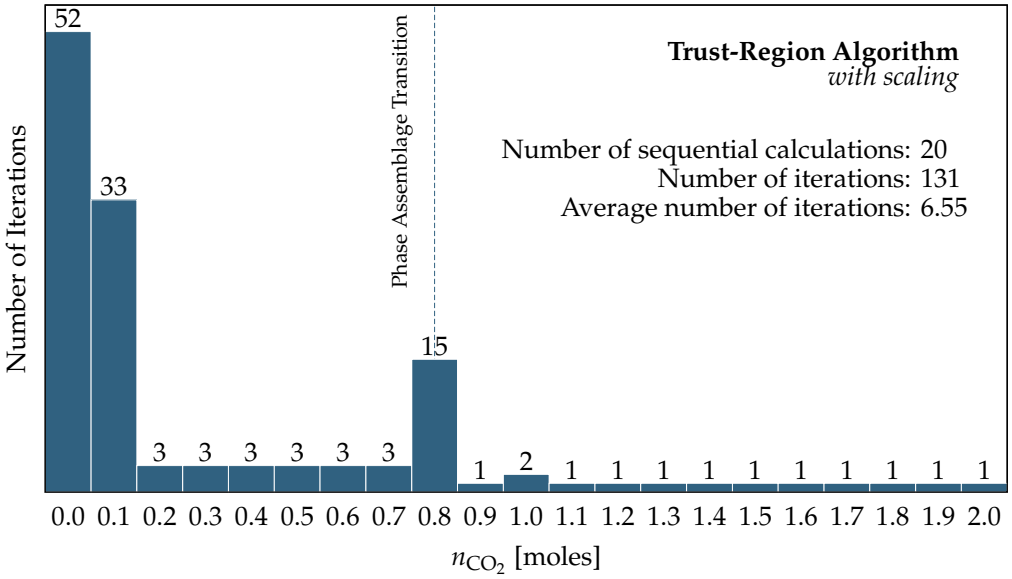


Figure 5.12: The performance of 20 sequential chemical equilibrium calculations given showing the number of iterations using the monotone trust-region algorithm.

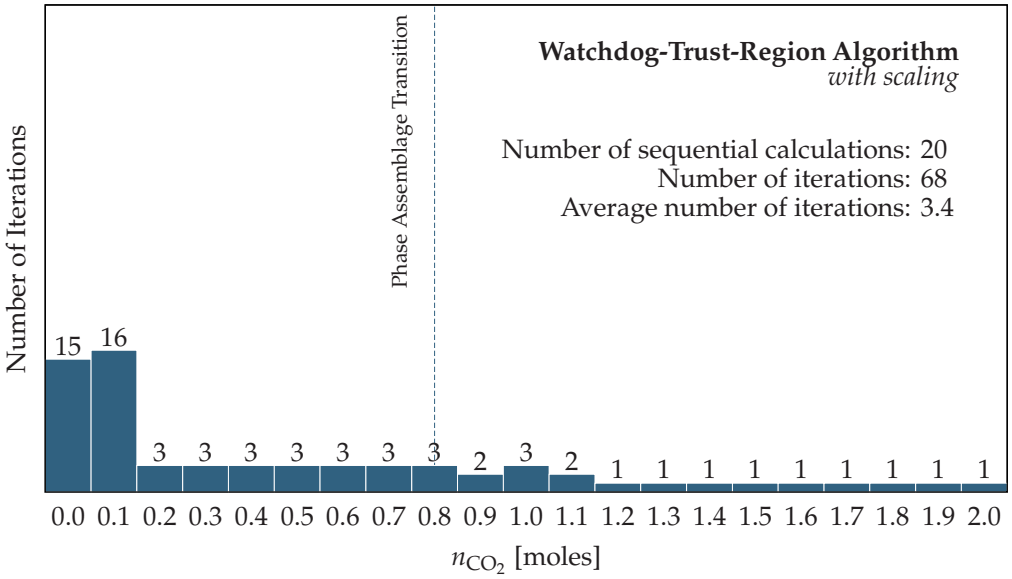


Figure 5.13: The performance of 20 sequential chemical equilibrium calculations showing the number of iterations using the non-monotone watchdog trust-region algorithm.

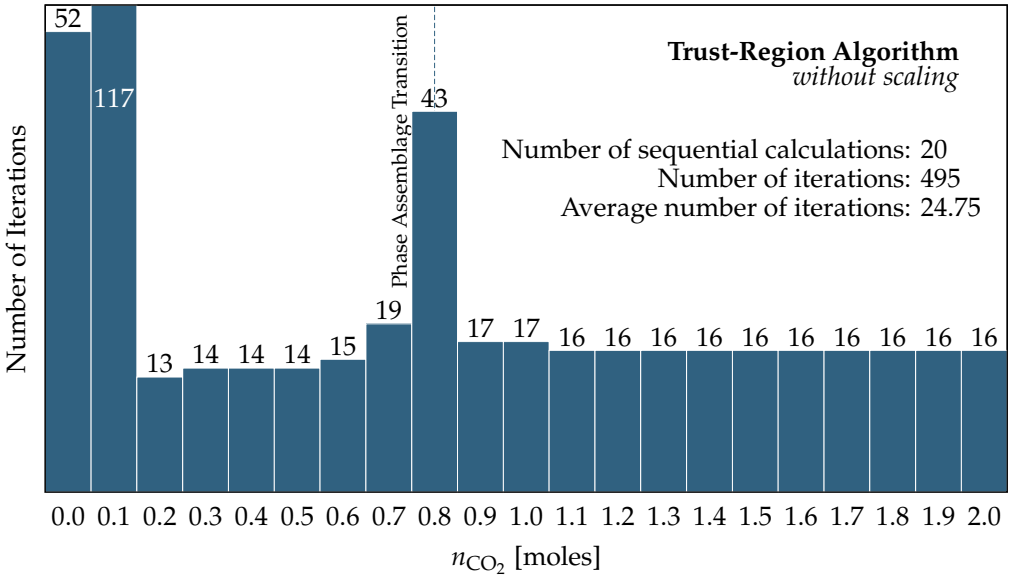


Figure 5.14: The performance of 20 sequential chemical equilibrium calculations showing the number of iterations using the monotone trust-region algorithm without scaling.

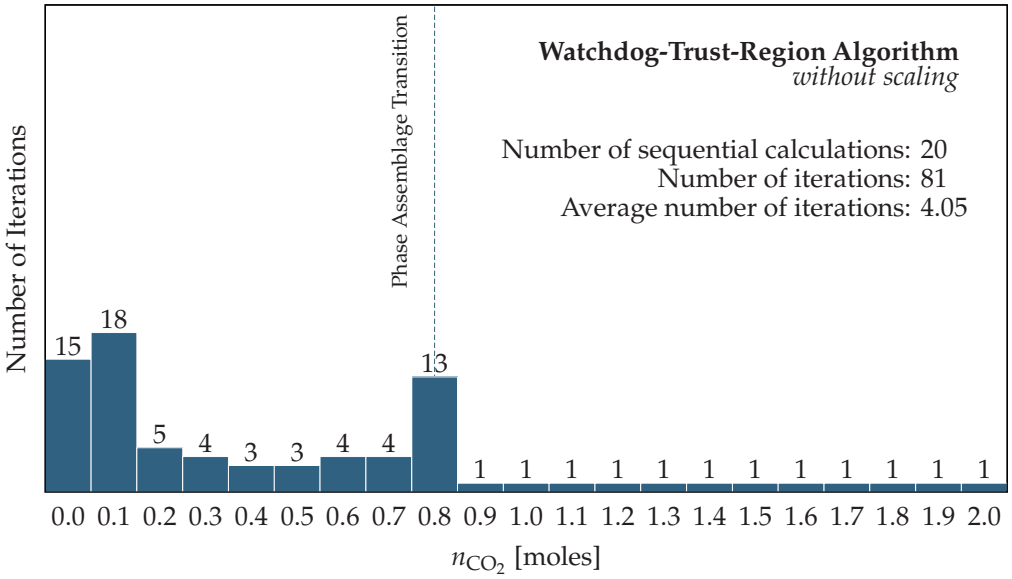


Figure 5.15: The performance of 20 sequential chemical equilibrium calculations showing the number of iterations using the non-monotone watchdog trust-region algorithm without scaling.

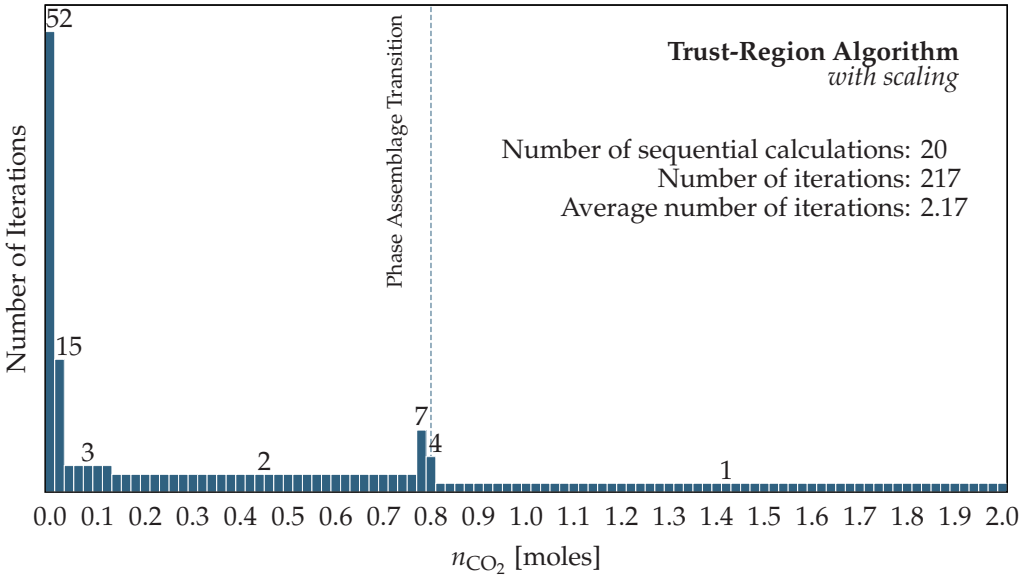


Figure 5.16: The performance of 100 sequential chemical equilibrium calculations showing the number of iterations using the monotone trust-region algorithm.

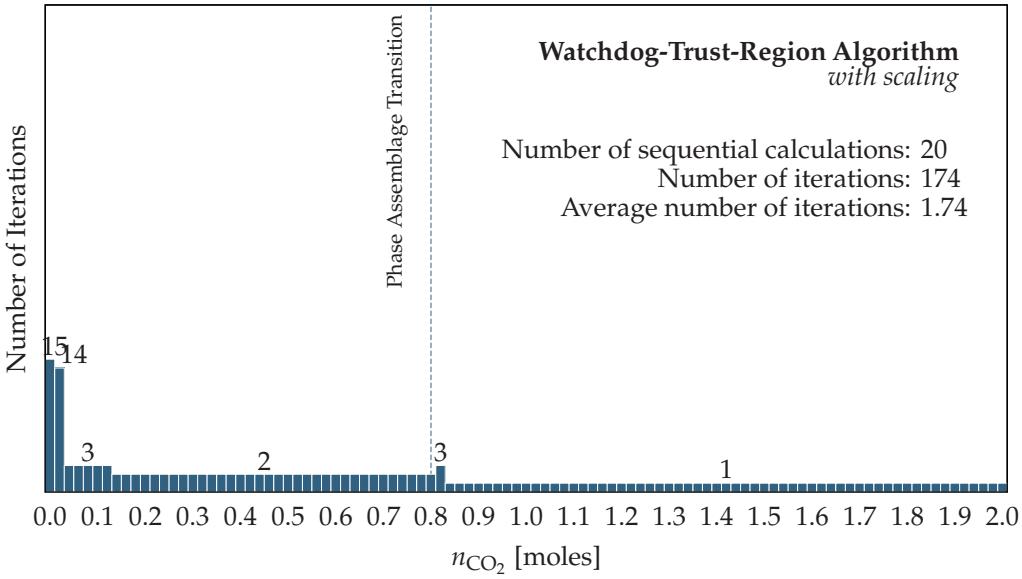


Figure 5.17: The performance of 100 sequential chemical equilibrium calculations showing the number of iterations using the non-monotone watchdog trust-region algorithm.

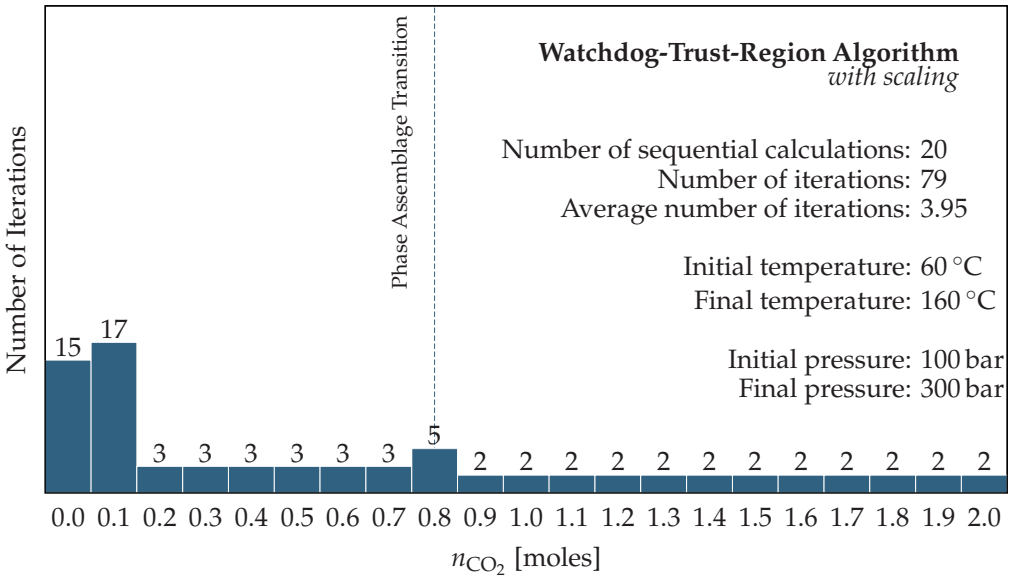


Figure 5.18: The performance of 20 sequential chemical equilibrium calculations showing the number of iterations using the non-monotone watchdog trust-region algorithm assuming variable temperatures and pressures.

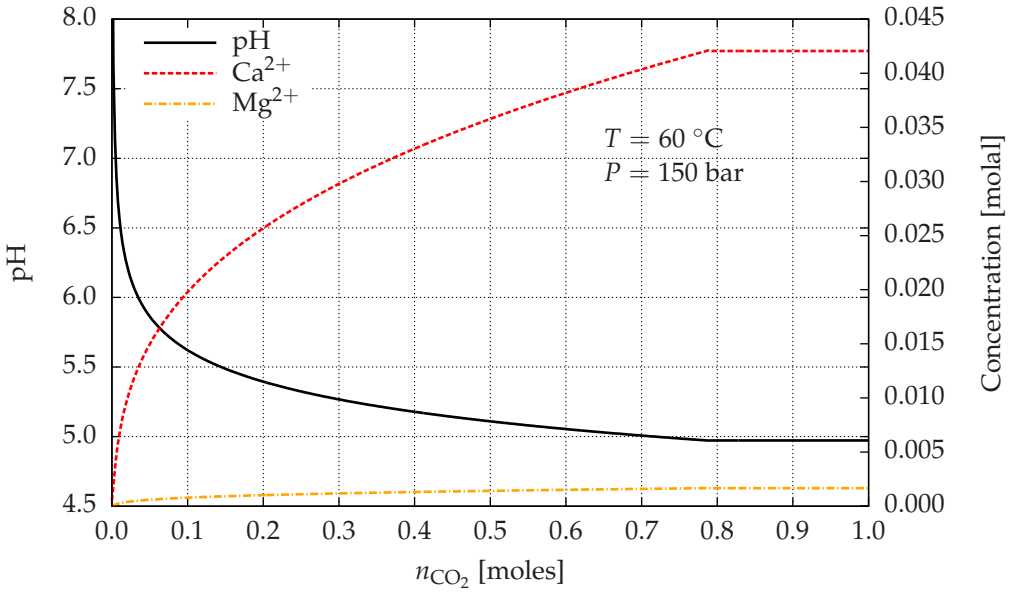


Figure 5.19: Calculated pH of the aqueous solution and the molalities of the ionic species Ca^{2+} and Mg^{2+} as the amount of CO_2 increases in the system at $T = 60\,°C$ and $P = 150\,bar$.

5.5 Convergence Rates of the Interior-Point Method

It was argued in Chapter 3 that superlinear rates of convergence are essential if this algorithm is to be incorporated in critical applications such as chemical kinetics and reactive transport modelling. Because initial guesses in these applications are usually close to the solution, with its proximity being mainly dependent on the used time step, it is of utmost importance that an equilibrium algorithm converges at fast rates locally. In this section we present results that permit us to analyse the convergence rates of the trust-region interior-point algorithm with and without the watchdog strategy.

Figures 5.20–5.22 present the convergence plot of equilibrium calculations assuming the multiphase system of Table 5.10. The residual of a calculation is defined as the Euclidean norm of the relaxed KKT function $\mathbf{F}_{\hat{\mu}}(\mathbf{w})$ given in equation (3.19). All calculations were performed using poor initial guesses, so that the initial residual would be large. The watchdog threshold $\mu_w = 10^{-6}$ was used instead of $\mu_w = 10^{-1}$ to create two distinct regions in the graphs that show when the watchdog strategy is activated. The solid circles indicate the activation of the watchdog strategy, and the empty circles indicate the maximum residual attained during the non-monotone iterations. Note that all calculations succeeded under the watchdog mode, without returning to the monotone trust-region strategy.

These calculations were performed with different equilibrium conditions. In Figure 5.20, for example, the amount of CO_2 in the system is not enough to produce a gaseous phase. In Figure 5.21, a gaseous phase is about to emerge, and so the calculation was performed near a phase boundary. In Figure 5.22, the specified amount of CO_2 was suficient to saturate the aqueous phase and to form a gaseous phase.

These results show that the use of the watchdog strategy boosts the convergence rate near the solution. This applies even in critical regions, such as those where a phase boundary exists. By using the non-monotone watchdog strategy, our calculations could achieve quadratic rates of convergence. These rates are superior to the linear ones obtained with the original monotone trust-region interior-point method of Ulbrich et al. (2004) and Silva et al. (2008), which can be explained by the poorly scaled minimisation problem and the lack of scaling invariance of this trust-region algorithm.

The convergence rates α_r of the calculations were computed using the formula:

$$\alpha_r := \log(r_{k+1}/r_k)/\log(r_k/r_{k-1}), \tag{5.5}$$

obtained by assuming that at the k-th iteration the following applies:

$$r_{k+1} = C r_k^{\alpha_r}, \tag{5.6}$$

where r_k denotes the residual at the k-th iteration, and C is a positive constant. The values of the convergence rates displayed in Figures 5.20–5.22 are average of the last three iterations.

5.6 Sensitivity of the Interior-Point Method

The description of the interior-point method in Chapter 3 did not contain any discussion concerning how sensitive its solutions are with respect to the final perturbation parameter $\hat{\mu}$. Recall

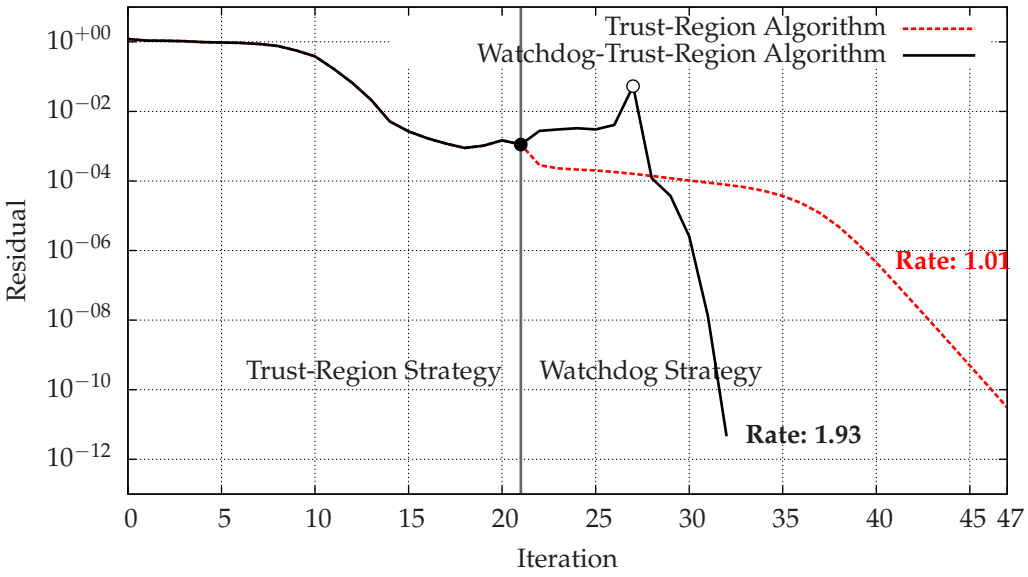


Figure 5.20: Residual of the equilibrium calculation at $n_{CO_2} = 0.1$ mol with and without the watchdog strategy. At this condition the gaseous phase is not present at equilibrium. The activation of the watchdog strategy is indicated by the solid circle, and the maximum residual attained during the non-monotone iterations is indicated by the empty circle.

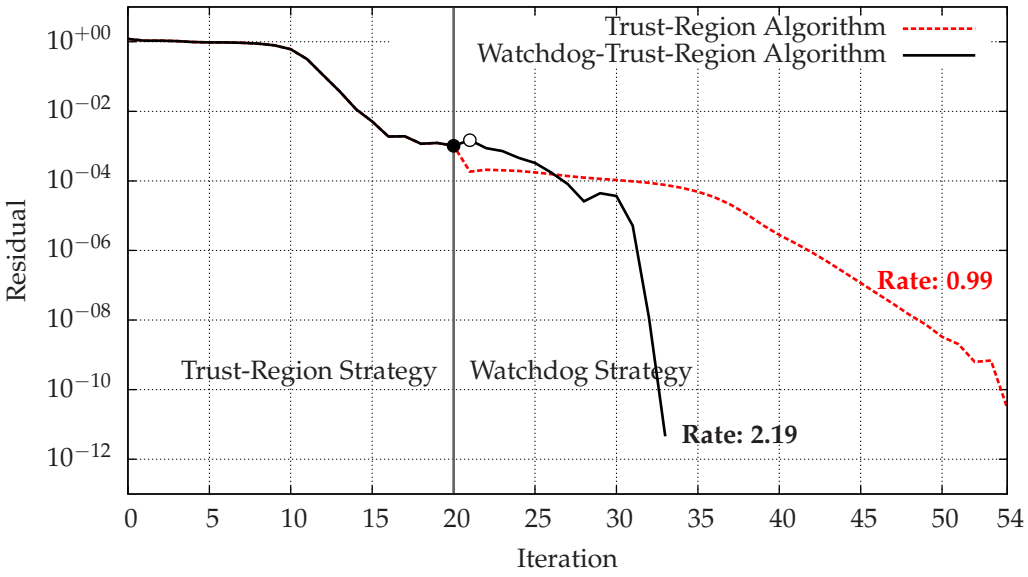


Figure 5.21: Residual of the equilibrium calculation at $n_{CO_2} = 0.793$ mol with and without the watchdog strategy. At this condition the gaseous phase is about to appear at equilibrium.

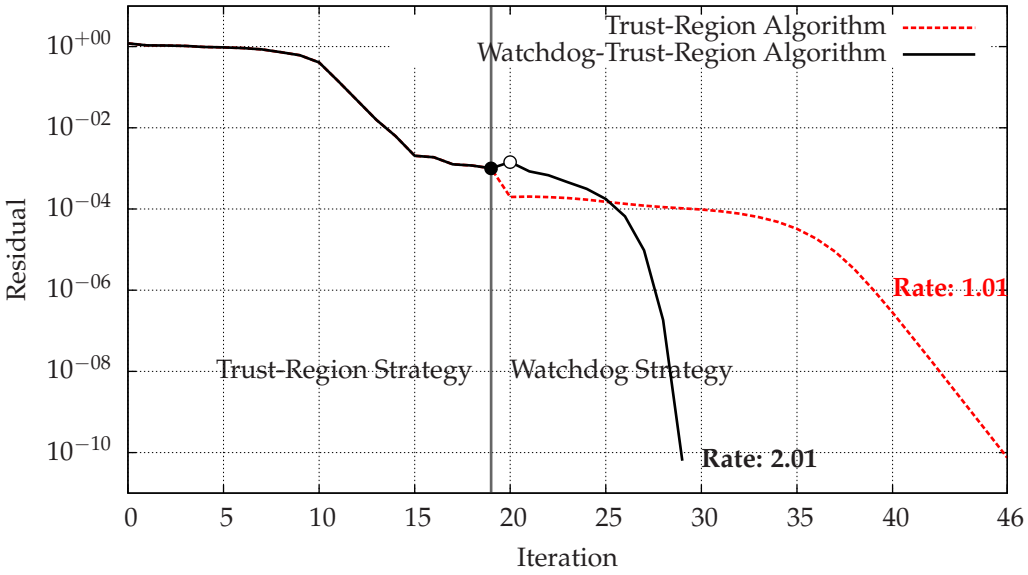


Figure 5.22: Residual of the equilibrium calculation at $n_{CO_2} = 1.0$ mol with and without the watchdog strategy. At this condition the gaseous phase is present at equilibrium.

that this parameter is used to relax the strict complementary conditions (3.15) of the KKT equations. Therefore, this relaxation is expected to perturb the solution as well, which we shall see in this section at which extent this happens. For this test the watchdog strategy is used with a small constant perturbation parameter $\dot{\mu}$ until convergence.

Consider the multiphase system of Table 5.14. Similarly as before, the molar amounts of components H_2O and NaCl will be fixed and CO_2 will be gradually added to the system. Eventually a gaseous phase will form, but before this happens we want to investigate the effect of parameter $\dot{\mu}$ on the number of moles of the unstable gaseous species. We should expect that these are tiny values in comparison with the number of moles in other phases.

Table 5.14: Description of the chemical system H_2O–CO_2–NaCl with their phases and respective chemical species.

Aqueous Phase	**Gaseous Phase**
$H_2O(l)$	$CO_2(g)$
H^+	$H_2O(g)$
OH^-	
Na^+	
Cl^-	
HCO_3^-	
$CO_2(aq)$	

Figure 5.23 shows the effect of different values of $\dot{\mu}$ on the number of moles of the gaseous species $CO_2(g)$. The figure shows that the smaller the perturbation parameter $\dot{\mu}$ is, the more accurate is the numerical representation of the inexistent gaseous species. For example, using $\dot{\mu} = 10^{-15}$ results in the inexistent $CO_2(g)$ having number of moles in the order of 10^{-13}. Compare this with the case $\dot{\mu} = 10^{-9}$, where the number of moles of $CO_2(g)$ in its inexistent region is about to 10^{-7}. Moreover, observe that the smaller the perturbation parameter $\dot{\mu}$ is, the sharper is the phase assemblage transition curve.

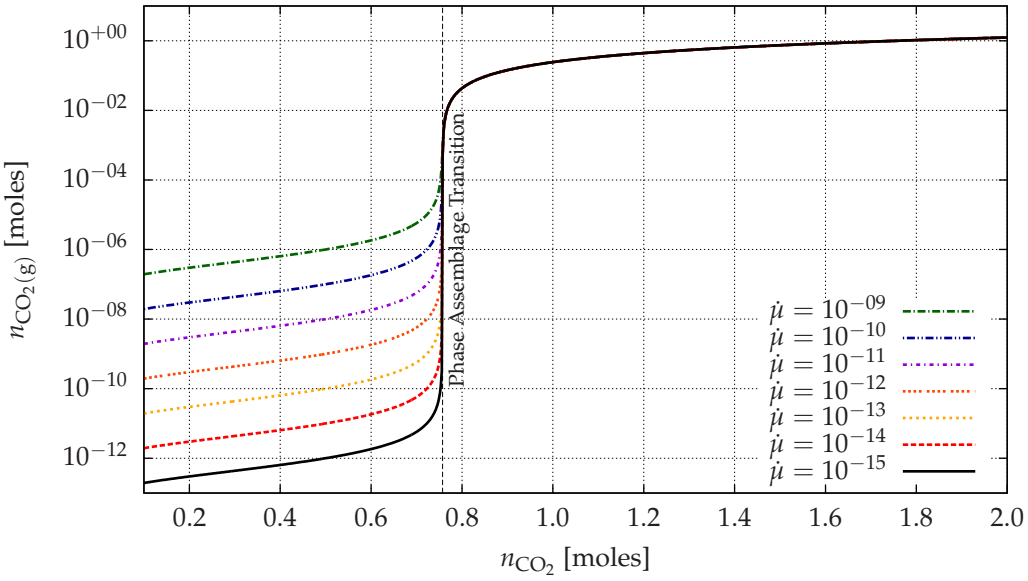


Figure 5.23: Effect of the perturbation parameter $\dot{\mu}$ on the number of moles of the gaseous species $CO_2(g)$, plotted in log-scale.

CHAPTER 6

APPLICATIONS: CHEMICAL KINETICS

In this chapter our chemical kinetics algorithm is applied to problems pertinent to carbon dioxide injection into saline aquifers.

The injection of carbon dioxide into saline aquifers perturbs the reservoir and initiates several physical and chemical phenomena due to the interactions of the injected gas with the resident fluid and the reservoir rock. The complexity of its computational modelling can be high when both transport processes (e.g., advection, diffusion, dispersion) and chemical processes (e.g., mineral dissolution/precipitation, gas dissolution/exsolution, etc.) are considered (see Pruess et al., 2003; Kumar et al., 2004; Ennis-King and Paterson, 2005, 2007; Xu et al., 2003, 2006; Obi and Blunt, 2006; Audigane et al., 2007).

Nevertheless, in order to apply the methodology presented in Chapter 4, all transport processes in this study are neglected. As mentioned before, it is assumed that the kinetic process occurs in a well-mixed batch reactor, although the method is designed for eventual incorporation in a reactive transport simulator.

6.1 Kinetic Modelling of CO_2 Injection into Carbonate Saline Aquifers

Consider a subsurface fluid in equilibrium with a carbonate rock. Assume that supercritical carbon dioxide is injected into this system with an amount large enough to saturate the fluid and produce a supercritical CO_2-rich phase. In order to model and analyse the water-gas-rock interactions produced by the gas injection, both chemical kinetics and chemical equilibrium methods presented earlier are used. The entire modelling can be divided into three stages:

Stage 1 calculation of the equilibrium state of the system comprised of the subsurface fluid and the rock-forming minerals;

Stage 2 calculation of the equilibrium state of the system comprised of the injected supercritical carbon dioxide and the resultant aqueous solution of the previous stage;

Stage 3 calculation of the transient state of the entire system comprised of the rock-forming minerals, the subsurface fluid, and the emerged CO_2-rich phase.

The first stage ensures that before carbon dioxide is injected, both fluid and rock are in thermodynamic equilibrium. In other words, it models the state of long residency time of the fluid in contact with the reservoir rock, as it would happen in a saline aquifer.

The second stage assumes that the injected carbon dioxide achieves equilibrium with the fluid considerably faster than the rock-forming minerals. This is a reasonable assumption since the speed of mineral dissolution is in general slower than the one for gas dissolution. Hence, we neglect any amount of mineral that is dissolved between the carbon dioxide injection and its equilibrium with the brine.

At the third stage, the rock-forming minerals are in disequilibrium with the rest of the system. By using the chemical kinetics methodology presented earlier, we can calculate the transient state of the entire system until it achieves equilibrium.

An important assumption in the previous modelling is that pressure is kept constant at all stages. In other words, we assume an expandable system that can accommodate the injected gas and permit an isobaric process. Support for a constant volume system is a planned work, which will allow the modelling of the pressure rise with gas injection.

6.1.1 Qatari Brine and Rock Composition

A brine composition representative of a Qatari subsurface fluid is considered. Table 6.2 presents the analysis of the subsurface fluid of two Qatari reservoirs. There are three samples for each reservoir, from which we can observe large differences in composition, even within the same reservoir. This lack of homogeneity motivated the choice of samples 2 and 3 of Reservoir B as the composition of the brines used in this study, which are denoted by Brine 1 and 2 respectively. Note that Brine 1 is the lower limit case in terms of concentrations of cations and anions, while Brine 2 is the upper limit case (about five times more concentrated than Brine 1). The presence of the components iron and barium is neglected in the calculations.

The mineral composition of the rock chosen in this work is also representative of a Qatari reservoir rock. Table 6.1 shows the volume percent of the minerals of two samples. The volume composition was obtained by X-ray diffraction analysis.

Table 6.1: The composition of a rock representative of a Qatari reservoir.

Mineral	Rock 1	Rock 2
Calcite	93.3%	97.2%
Dolomite	5.2%	0.0%
Quartz	1.5%	2.8%

Note: Composition in units of volume percent.

Given the composition of the subsurface fluid and rock, it is now necessary to define the multiphase chemical system to be used in the computational modelling. Using the database of Johnson et al. (1992), SUPCRT, all possible aqueous species that could be present in the subsurface fluid are determined. These are listed in Table 6.3, which also shows the assumed gaseous and mineral species. The pure mineral phases composed of magnesite and halite are considered to capture eventual secondary mineral precipitation.

Table 6.2: The subsurface fluid composition of two Qatari reservoirs.

	Reservoir A			Reservoir B		
Composition	**Sample 1**	**Sample 2**	**Sample 3**	**Sample 1**	**Sample 2 Brine 1**	**Sample 3 Brine 2**
Cations						
Na^+	8320	9930	14177	14204	**7180**	**33838**
Ca^{2+}	1483	1200	3240	5109	**1500**	**10586**
Mg^{2+}	316	425	624	1128	**345**	**1851**
K^+	1574	365	484	800	**300**	**460**
Sr^{2+}	40	63	98	159	**48**	**299**
Fe^{2+}, Fe^{3+}	< 1	—	0.4	2	**—**	**3**
Ba^{2+}	—	< 0.05	—	—	**0.23**	**—**
Anions						
Cl^-	16632	17880	28795	33969	**13660**	**76113**
HCO_3^-	315	475	277	258	**390**	**170**
SO_4^{2-}	1340	1230	1194	1147	**1520**	**1015**
CO_3^{2-}	0	0	0	0	**32**	**0**
OH^-	0	0	0	0	**0**	**0**

Note: Composition in units of mg/l, where l is volume of solvent in litres. The two samples labelled Brine 1 and Brine 2 are used for quantitative analysis.

Table 6.3: The chemical system for the representation of the subsurface fluid and rock of a Qatari reservoir.

Aqueous Phase					
$Ca(HCO_3)^+$	$CO(aq)$	HCO_3^-	KSO_4^-	$O_2(aq)$	$S_5O_6^{2-}$
$Ca(HSiO_3)^+$	$CO_2(aq)$	HO_2^-	$Mg(HCO_3)^+$	OH^-	$SiO_2(aq)$
Ca^{2+}	CO_3^{2-}	HS^-	$Mg(HSiO_3)^+$	S_2^{2-}	$SO_2(aq)$
$CaCl^+$	H^+	$HS_2O_3^-$	Mg^{2+}	$S_2O_3^{2-}$	SO_3^{2-}
$CaCl_2(aq)$	$H_2(aq)$	$HS_2O_4^-$	$MgCl^+$	$S_2O_4^{2-}$	SO_4^{2-}
$CaCO_3(aq)$	$H_2O(l)$	$HSiO_3^-$	$MgCO_3(aq)$	$S_2O_5^{2-}$	$Sr(HCO_3)^+$
$CaOH^+$	$H_2O_2(aq)$	HSO_3^-	$MgOH^+$	$S_2O_6^{2-}$	Sr^{2+}
$CaSO_4(aq)$	$H_2S(aq)$	HSO_4^-	$MgSO_4(aq)$	$S_2O_8^{2-}$	$SrCl^+$
Cl^-	$H_2S_2O_3(aq)$	HSO_5^-	Na^+	S_3^{2-}	$SrCO_3(aq)$
ClO^-	$H_2S_2O_4(aq)$	K^+	$NaCl(aq)$	$S_3O_6^{2-}$	$SrOH^+$
ClO_4^-	$HClO_2(aq)$	$KOH(aq)$	$NaSO_4^-$	S_5^{2-}	
ClO_3^-	$HClO(aq)$	$KHSO_4(aq)$	$NaOH(aq)$	$S_4O_6^{2-}$	
ClO_2^-	$HCl(aq)$	$KCl(aq)$	$NaHSiO_3(aq)$	S_4^{2-}	
Gaseous Phase					
$H_2O(g)$	$CO_2(g)$				
Mineral Phases					
Calcite	Dolomite	Quartz	Magnesite	Halite	

The chemical system in Table 6.3 contains several aqueous species. Table 6.4 shows the result of the equilibrium calculation at Stage 1, which corresponds to the equilibrium state of the subsurface fluid (Brine 1) and carbonate rock (Rock 1) at 60 °C and 150 bar. Note that several aqueous species are present only at very low concentrations. As a result, one can argue that many of these species could be potentially removed without compromising the accuracy of the calculation. This would also dramatically improve efficiency of the calculations, which is specially important for reactive flow simulations due to the large number of chemical equilibrium and kinetics calculations.

However, this must be done very carefully. For example, a species that is present at low concentrations initially can later increase considerably during the kinetics process. In this case, the calculation of the evolution of the simplified system could be inaccurate or even become unstable. Therefore, the full chemical system described in Table 6.3 was used in this work.

After determining the preliminary equilibrium states in Stage 1 and 2, it is then time to calculate the desired transient state of the multiphase system in Stage 3. As pointed out before, at the end of Stage 2, the rock-forming minerals are in disequilibrium with the CO_2 saturated subsurface fluid, which leads to a kinetic process of mineral dissolution/precipitation. In what follows, we will show the kinetic changes that occur in the chemical system until it achieves equilibrium.

The calculations assumed a subsurface fluid with a solvent mass of 1 kg and composition given in Table 6.2; a subsurface rock sample of 500 cm^3 with porosity 0.30, and composition given in Table 6.1; and 440 g of supercritical CO_2. The mineral rate parameters compiled by Palandri and Kharaka (2004) were used in the calculations. The data they present was compiled from several sources, and it has been used extensively in the literature for modelling carbon dioxide storage in saline aquifers, and the quantification of its trapping by mineral mechanisms. In addition, it is also adopted by the TOUGHREACT simulator (see Gunter et al., 2004; Xu et al., 2006, 2007; André et al., 2007; Gaus et al., 2008).

Table 6.5 shows the specific surface areas of the minerals controlled by kinetics at Stage 3 of the modelling problem. The choice of the specific surface area of calcite was motivated from the discussion in Schultz et al. (2013) and the value used in Garcia et al. (2011) for modelling calcite dissolution during geological CO_2 sequestration. The specific surface area of quartz was chosen based on the range of values determined by Leamnson et al. (1969) with the Brunauer-Emmet-Teller (BET) gas adsorption method. For the other carbonate minerals, the choice of specific surface area was an intermediate value between the ones chosen for calcite and quartz.

Table 6.5: The specific surface area of the rock minerals.

Mineral	**SSA** [m^2/g]
Calcite	1.5
Dolomite	1.0
Quartz	0.5
Magnesite	0.8

6.1.2 Results

Figure 6.1 shows the transient state of mineral calcite in Rocks 1 and 2 reacting with Brines 1 and 2 for one month. Observe that Rock 1 achieves equilibrium with the subsurface fluid and the CO_2-rich phase in a few minutes, while Rock 2 requires just a few seconds. It takes longer

Table 6.4: The chemical state of the system at the end of Stage 1, assuming Brine 1 and Rock 1.

Species	Amount [mol]	Activity	Activity Coefficient	Concentration*
Aqueous Phase				
$H_2O(l)$	$5.5510 \cdot 10^{+1}$	$9.8751 \cdot 10^{-1}$	$9.8751 \cdot 10^{-1}$	$9.8656 \cdot 10^{-1}$
Cl^-	$3.7261 \cdot 10^{-1}$	$2.5195 \cdot 10^{-1}$	$6.7618 \cdot 10^{-1}$	$3.7260 \cdot 10^{-1}$
Na^+	$2.9887 \cdot 10^{-1}$	$2.0064 \cdot 10^{-1}$	$6.7135 \cdot 10^{-1}$	$2.9886 \cdot 10^{-1}$
Ca^{2+}	$2.2505 \cdot 10^{-2}$	$4.2696 \cdot 10^{-3}$	$1.8972 \cdot 10^{-1}$	$2.2505 \cdot 10^{-2}$
Mg^{2+}	$2.1171 \cdot 10^{-2}$	$4.0145 \cdot 10^{-3}$	$1.8962 \cdot 10^{-1}$	$2.1171 \cdot 10^{-2}$
$NaCl(aq)$	$9.9797 \cdot 10^{-3}$	$1.0923 \cdot 10^{-2}$	$1.0946 \cdot 10^{+0}$	$9.9795 \cdot 10^{-3}$
SO_4^{2-}	$9.2728 \cdot 10^{-3}$	$1.8152 \cdot 10^{-3}$	$1.9576 \cdot 10^{-1}$	$9.2726 \cdot 10^{-3}$
K^+	$7.5342 \cdot 10^{-3}$	$5.0553 \cdot 10^{-3}$	$6.7099 \cdot 10^{-1}$	$7.5340 \cdot 10^{-3}$
$NaSO_4^-$	$3.4605 \cdot 10^{-3}$	$2.3352 \cdot 10^{-3}$	$6.7485 \cdot 10^{-1}$	$3.4604 \cdot 10^{-3}$
HCO_3^-	$2.8933 \cdot 10^{-3}$	$1.9512 \cdot 10^{-3}$	$6.7441 \cdot 10^{-1}$	$2.8932 \cdot 10^{-3}$
$MgSO_4(aq)$	$1.5802 \cdot 10^{-3}$	$1.7295 \cdot 10^{-3}$	$1.0946 \cdot 10^{+0}$	$1.5801 \cdot 10^{-3}$
$MgCl^+$	$1.3062 \cdot 10^{-3}$	$9.0492 \cdot 10^{-4}$	$6.9282 \cdot 10^{-1}$	$1.3062 \cdot 10^{-3}$
$CaSO_4(aq)$	$1.2585 \cdot 10^{-3}$	$1.3774 \cdot 10^{-3}$	$1.0946 \cdot 10^{+0}$	$1.2584 \cdot 10^{-3}$
$CaCl^+$	$1.2583 \cdot 10^{-3}$	$8.4957 \cdot 10^{-4}$	$6.7521 \cdot 10^{-1}$	$1.2582 \cdot 10^{-3}$
$CO_2(aq)$	$9.9510 \cdot 10^{-4}$	$1.0843 \cdot 10^{-3}$	$1.0897 \cdot 10^{+0}$	$9.9508 \cdot 10^{-4}$
Sr^{2+}	$5.0368 \cdot 10^{-4}$	$9.5775 \cdot 10^{-5}$	$1.9016 \cdot 10^{-1}$	$5.0367 \cdot 10^{-4}$
$SiO_2(aq)$	$3.1843 \cdot 10^{-4}$	$3.4853 \cdot 10^{-4}$	$1.0946 \cdot 10^{+0}$	$3.1842 \cdot 10^{-4}$
$Ca(HCO_3)^+$	$1.7564 \cdot 10^{-4}$	$1.1786 \cdot 10^{-4}$	$6.7107 \cdot 10^{-1}$	$1.7563 \cdot 10^{-4}$
$Mg(HCO_3)^+$	$1.6373 \cdot 10^{-4}$	$1.1128 \cdot 10^{-4}$	$6.7966 \cdot 10^{-1}$	$1.6372 \cdot 10^{-4}$
KSO_4^-	$1.3125 \cdot 10^{-4}$	$8.9220 \cdot 10^{-5}$	$6.7977 \cdot 10^{-1}$	$1.3125 \cdot 10^{-4}$
HS^-	$6.7359 \cdot 10^{-5}$	$4.5523 \cdot 10^{-5}$	$6.7585 \cdot 10^{-1}$	$6.7357 \cdot 10^{-5}$
$H_2S(aq)$	$5.7890 \cdot 10^{-5}$	$6.3362 \cdot 10^{-5}$	$1.0946 \cdot 10^{+0}$	$5.7888 \cdot 10^{-5}$
$CaCl_2(aq)$	$5.3989 \cdot 10^{-5}$	$5.9093 \cdot 10^{-5}$	$1.0946 \cdot 10^{+0}$	$5.3988 \cdot 10^{-5}$
$SrCl^+$	$3.1453 \cdot 10^{-5}$	$2.1146 \cdot 10^{-5}$	$6.7232 \cdot 10^{-1}$	$3.1452 \cdot 10^{-5}$
$Sr(HCO_3)^+$	$1.3031 \cdot 10^{-5}$	$8.8600 \cdot 10^{-6}$	$6.7993 \cdot 10^{-1}$	$1.3031 \cdot 10^{-5}$
$CaCO_3(aq)$	$9.0122 \cdot 10^{-6}$	$9.8641 \cdot 10^{-6}$	$1.0946 \cdot 10^{+0}$	$9.0120 \cdot 10^{-6}$
$Ca(CO_3)(aq)$	$8.9358 \cdot 10^{-6}$	$9.7805 \cdot 10^{-6}$	$1.0946 \cdot 10^{+0}$	$8.9356 \cdot 10^{-6}$
$KCl(aq)$	$7.9128 \cdot 10^{-6}$	$8.6607 \cdot 10^{-6}$	$1.0946 \cdot 10^{+0}$	$7.9126 \cdot 10^{-6}$
$NaHSiO_3(aq)$	$3.5817 \cdot 10^{-6}$	$3.9202 \cdot 10^{-6}$	$1.0946 \cdot 10^{+0}$	$3.5816 \cdot 10^{-6}$
$MgCO_3(aq)$	$2.8104 \cdot 10^{-6}$	$3.0761 \cdot 10^{-6}$	$1.0946 \cdot 10^{+0}$	$2.8103 \cdot 10^{-6}$
$Mg(CO_3)(aq)$	$2.7854 \cdot 10^{-6}$	$3.0487 \cdot 10^{-6}$	$1.0946 \cdot 10^{+0}$	$2.7853 \cdot 10^{-6}$
CO_3^{2-}	$2.5987 \cdot 10^{-6}$	$5.0597 \cdot 10^{-7}$	$1.9471 \cdot 10^{-1}$	$2.5986 \cdot 10^{-6}$
$HSiO_3^-$	$1.0065 \cdot 10^{-6}$	$6.8335 \cdot 10^{-7}$	$6.7896 \cdot 10^{-1}$	$1.0065 \cdot 10^{-6}$
$MgOH^+$	$5.2361 \cdot 10^{-7}$	$3.6276 \cdot 10^{-7}$	$6.9282 \cdot 10^{-1}$	$5.2359 \cdot 10^{-7}$
H^+	$4.9279 \cdot 10^{-7}$	$3.3418 \cdot 10^{-7}$	$6.7816 \cdot 10^{-1}$	$4.9277 \cdot 10^{-7}$
OH^-	$4.4893 \cdot 10^{-7}$	$3.0813 \cdot 10^{-7}$	$6.8638 \cdot 10^{-1}$	$4.4892 \cdot 10^{-7}$
HSO_4^-	$2.2339 \cdot 10^{-7}$	$1.5076 \cdot 10^{-7}$	$6.7489 \cdot 10^{-1}$	$2.2338 \cdot 10^{-7}$
$SrCO_3(aq)$	$7.7890 \cdot 10^{-8}$	$8.5252 \cdot 10^{-8}$	$1.0946 \cdot 10^{+0}$	$7.7888 \cdot 10^{-8}$
$Sr(CO_3)(aq)$	$7.7212 \cdot 10^{-8}$	$8.4511 \cdot 10^{-8}$	$1.0946 \cdot 10^{+0}$	$7.7211 \cdot 10^{-8}$
$Mg(HSiO_3)^+$	$7.6408 \cdot 10^{-8}$	$5.3287 \cdot 10^{-8}$	$6.9743 \cdot 10^{-1}$	$7.6406 \cdot 10^{-8}$
$CaOH^+$	$6.9268 \cdot 10^{-8}$	$4.6688 \cdot 10^{-8}$	$6.7404 \cdot 10^{-1}$	$6.9266 \cdot 10^{-8}$
$Ca(HSiO_3)^+$	$5.6223 \cdot 10^{-8}$	$3.8173 \cdot 10^{-8}$	$6.7897 \cdot 10^{-1}$	$5.6222 \cdot 10^{-8}$
$NaOH(aq)$	$3.5695 \cdot 10^{-8}$	$3.9069 \cdot 10^{-8}$	$1.0946 \cdot 10^{+0}$	$3.5694 \cdot 10^{-8}$
$HCl(aq)$	$1.3607 \cdot 10^{-8}$	$1.4893 \cdot 10^{-8}$	$1.0946 \cdot 10^{+0}$	$1.3606 \cdot 10^{-8}$
$S_2O_3^{2-}$	$5.3945 \cdot 10^{-9}$	$1.0681 \cdot 10^{-9}$	$1.9801 \cdot 10^{-1}$	$5.3944 \cdot 10^{-9}$
$KOH(aq)$	$6.9884 \cdot 10^{-10}$	$7.6490 \cdot 10^{-10}$	$1.0946 \cdot 10^{+0}$	$6.9883 \cdot 10^{-10}$
$SrOH^+$	$6.3884 \cdot 10^{-10}$	$4.2888 \cdot 10^{-10}$	$6.7135 \cdot 10^{-1}$	$6.3882 \cdot 10^{-10}$
$H_2(aq)$	$5.2701 \cdot 10^{-10}$	$5.7682 \cdot 10^{-10}$	$1.0946 \cdot 10^{+0}$	$5.2700 \cdot 10^{-10}$
S_2^{2-}	$3.2717 \cdot 10^{-11}$	$6.4255 \cdot 10^{-12}$	$1.9640 \cdot 10^{-1}$	$3.2716 \cdot 10^{-11}$
HSO_3^-	$2.8853 \cdot 10^{-12}$	$1.9577 \cdot 10^{-12}$	$6.7850 \cdot 10^{-1}$	$2.8853 \cdot 10^{-12}$
S_3^{2-}	$1.4243 \cdot 10^{-12}$	$2.8191 \cdot 10^{-13}$	$1.9793 \cdot 10^{-1}$	$1.4243 \cdot 10^{-12}$
SO_3^{2-}	$1.2781 \cdot 10^{-12}$	$2.4924 \cdot 10^{-13}$	$1.9502 \cdot 10^{-1}$	$1.2781 \cdot 10^{-12}$

Table 6.4: Continued from previous page

Species	**Amount** [mol]	**Activity**	**Activity Coefficient**	**Concentration***
$KHSO_4(aq)$	$6.2171 \cdot 10^{-13}$	$6.8047 \cdot 10^{-13}$	$1.0946 \cdot 10^{+0}$	$6.2169 \cdot 10^{-13}$
$HS_2O_3^-$	$4.3823 \cdot 10^{-14}$	$2.9653 \cdot 10^{-14}$	$6.7665 \cdot 10^{-1}$	$4.3822 \cdot 10^{-14}$
S_4^{2-}	$3.9098 \cdot 10^{-14}$	$7.8247 \cdot 10^{-15}$	$2.0013 \cdot 10^{-1}$	$3.9098 \cdot 10^{-14}$
$CO(aq)$	$1.0235 \cdot 10^{-14}$	$1.1203 \cdot 10^{-14}$	$1.0946 \cdot 10^{+0}$	$1.0235 \cdot 10^{-14}$
S_5^{2-}	$6.7962 \cdot 10^{-16}$	$1.3800 \cdot 10^{-16}$	$2.0307 \cdot 10^{-1}$	$6.7960 \cdot 10^{-16}$
$SO_2(aq)$	$9.6133 \cdot 10^{-17}$	$1.0522 \cdot 10^{-16}$	$1.0946 \cdot 10^{+0}$	$9.6131 \cdot 10^{-17}$
$H_2S_2O_3(aq)$	$6.5028 \cdot 10^{-20}$	$7.1174 \cdot 10^{-20}$	$1.0946 \cdot 10^{+0}$	$6.5026 \cdot 10^{-20}$
$S_4O_6^{2-}$	$4.0292 \cdot 10^{-24}$	$8.8564 \cdot 10^{-25}$	$2.1981 \cdot 10^{-1}$	$4.0291 \cdot 10^{-24}$
$S_2O_4^{2-}$	$1.9482 \cdot 10^{-27}$	$3.8854 \cdot 10^{-28}$	$1.9944 \cdot 10^{-1}$	$1.9481 \cdot 10^{-27}$
$S_2O_5^{2-}$	$2.5317 \cdot 10^{-28}$	$5.0690 \cdot 10^{-29}$	$2.0022 \cdot 10^{-1}$	$2.5317 \cdot 10^{-28}$
$S_2O_6^{2-}$	$1.8348 \cdot 10^{-31}$	$3.7024 \cdot 10^{-32}$	$2.0179 \cdot 10^{-1}$	$1.8348 \cdot 10^{-31}$
$HS_2O_4^-$	$7.8631 \cdot 10^{-32}$	$5.3567 \cdot 10^{-32}$	$6.8126 \cdot 10^{-1}$	$7.8629 \cdot 10^{-32}$
$S_3O_6^{2-}$	$3.5191 \cdot 10^{-35}$	$7.1389 \cdot 10^{-36}$	$2.0287 \cdot 10^{-1}$	$3.5190 \cdot 10^{-35}$
$H_2S_2O_4(aq)$	$7.5244 \cdot 10^{-38}$	$8.2356 \cdot 10^{-38}$	$1.0946 \cdot 10^{+0}$	$7.5242 \cdot 10^{-38}$
$S_5O_6^{2-}$	$5.7121 \cdot 10^{-40}$	$1.1754 \cdot 10^{-40}$	$2.0578 \cdot 10^{-1}$	$5.7120 \cdot 10^{-40}$
$HClO(aq)$	$3.3927 \cdot 10^{-46}$	$3.7134 \cdot 10^{-46}$	$1.0946 \cdot 10^{+0}$	$3.3926 \cdot 10^{-46}$
ClO^-	$8.5263 \cdot 10^{-47}$	$5.7696 \cdot 10^{-47}$	$6.7670 \cdot 10^{-1}$	$8.5261 \cdot 10^{-47}$
$H_2O_2(aq)$	$2.2185 \cdot 10^{-47}$	$2.4282 \cdot 10^{-47}$	$1.0946 \cdot 10^{+0}$	$2.2184 \cdot 10^{-47}$
HO_2^-	$8.4585 \cdot 10^{-52}$	$5.7410 \cdot 10^{-52}$	$6.7875 \cdot 10^{-1}$	$8.4583 \cdot 10^{-52}$
HSO_5^-	$2.9975 \cdot 10^{-56}$	$2.1271 \cdot 10^{-56}$	$7.0965 \cdot 10^{-1}$	$2.9974 \cdot 10^{-56}$
$O_2(aq)$	$2.4063 \cdot 10^{-64}$	$2.6338 \cdot 10^{-64}$	$1.0946 \cdot 10^{+0}$	$2.4063 \cdot 10^{-64}$
$S_2O_8^{2-}$	$3.3290 \cdot 10^{-69}$	$7.2295 \cdot 10^{-70}$	$2.1717 \cdot 10^{-1}$	$3.3289 \cdot 10^{-69}$
ClO_2^-	$9.2042 \cdot 10^{-86}$	$6.2089 \cdot 10^{-86}$	$6.7459 \cdot 10^{-1}$	$9.2040 \cdot 10^{-86}$
$HClO_2(aq)$	$3.4061 \cdot 10^{-90}$	$3.7280 \cdot 10^{-90}$	$1.0946 \cdot 10^{+0}$	$3.4060 \cdot 10^{-90}$
ClO_3^-	$2.5781 \cdot 10^{-112}$	$1.7630 \cdot 10^{-112}$	$6.8384 \cdot 10^{-1}$	$2.5780 \cdot 10^{-112}$
ClO_4^-	$6.3981 \cdot 10^{-143}$	$4.3996 \cdot 10^{-143}$	$6.8766 \cdot 10^{-1}$	$6.3979 \cdot 10^{-143}$
Gaseous Phase				
$H_2O(g)$	$0.0000 \cdot 10^{+0}$	$0.0000 \cdot 10^{+0}$	$1.0000 \cdot 10^{+0}$	$0.0000 \cdot 10^{+0}$
$CO_2(g)$	$0.0000 \cdot 10^{+0}$	$0.0000 \cdot 10^{+0}$	$1.0000 \cdot 10^{+0}$	$0.0000 \cdot 10^{+0}$
Mineral Phases				
Calcite	$8.8636 \cdot 10^{+0}$	$1.0000 \cdot 10^{+0}$	$1.0000 \cdot 10^{+0}$	$1.0000 \cdot 10^{+0}$
Dolomite	$2.7262 \cdot 10^{-1}$	$1.0000 \cdot 10^{+0}$	$1.0000 \cdot 10^{+0}$	$1.0000 \cdot 10^{+0}$
Quartz	$2.3108 \cdot 10^{-1}$	$1.0000 \cdot 10^{+0}$	$1.0000 \cdot 10^{+0}$	$1.0000 \cdot 10^{+0}$
Halite	$0.0000 \cdot 10^{+0}$	$0.0000 \cdot 10^{+0}$	$1.0000 \cdot 10^{+0}$	$0.0000 \cdot 10^{+0}$
Magnesite	$0.0000 \cdot 10^{+0}$	$0.0000 \cdot 10^{+0}$	$1.0000 \cdot 10^{+0}$	$0.0000 \cdot 10^{+0}$

Note: Concentration in units of molality for aqueous solutes, and molar fractions for all other species.

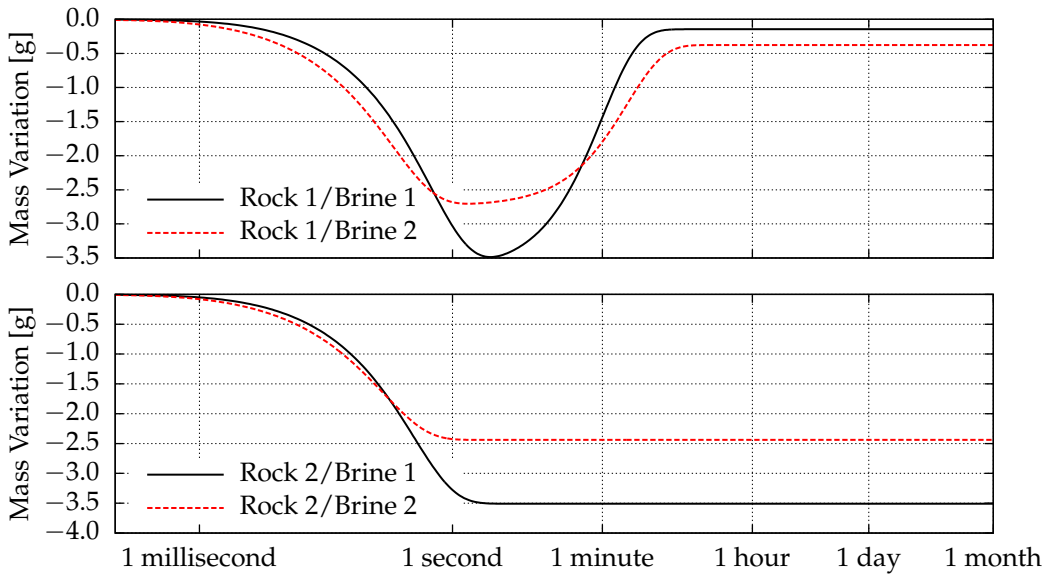


Figure 6.1: The transient state of mineral calcite during one month of reaction with the CO_2 saturated subsurface fluid.

for Rock 1 to achieve equilibrium because it has a period of *calcite precipitation* that starts after a few seconds, which does not happen with Rock 2. Recall that Rock 1 has dolomite in its composition, which releases ions Ca^{2+} during dissolution that can react with the free bicarbonate ions HCO_3^- in the subsurface fluid to precipitate calcite. This explains why precipitation of calcite is not seen with the dolomite-free Rock 2. Note that the precipitation of calcite in Rock 1 is almost enough to compensate for the initial dissolution.

Figure 6.2 shows the transient dissolution of dolomite in Rock 1 with Brines 1 and 2. A salting-out effect can be seen in the dissolution of dolomite, where Brine 2 dissolves less because it is more saline than Brine 1. When assuming the dolomite-free Rock 2 in the system, there was no dolomite precipitation for both brines. Thus, the subsurface fluid samples are undersaturated with ions Ca^{2+} and Mg^{2+}.

Figure 6.3 presents the transient state of the pH of the subsurface fluids for the two rocks and brines. Observe that the dissolution of the carbonate minerals is followed by an increase in the pH of the aqueous solution. At both initial and final times, the solution is more acidic for Brine 2. However, for a short time during the dissolution of the minerals, it can be seen that the two curves overlap or almost intercept each other. This shows the complex non-linear behaviour of the system with respect to salinity.

In Figure 6.4 we see the concentration evolution of the ionic species Ca^{2+}, Mg^{2+} and HCO_3^-, where the chemical system was composed of Rock 1 and Brine 1. Between 1 second and 1 minute there is a slowdown in the increase of HCO_3^-, while at the same time the concentration of Ca^{2+} is decreasing, suggesting that HCO_3^- and Ca^{2+} are reacting to produce calcite (see Figure 6.1). Note that the concentration of HCO_3^- still increases during this period, which means that its production from the dissolution of supercritical CO_2 is higher than its consumption by the precipitation of calcite. The concentration of the ionic species Mg^{2+} increases at the same proportion as dolomite is dissolved into the aqueous solution.

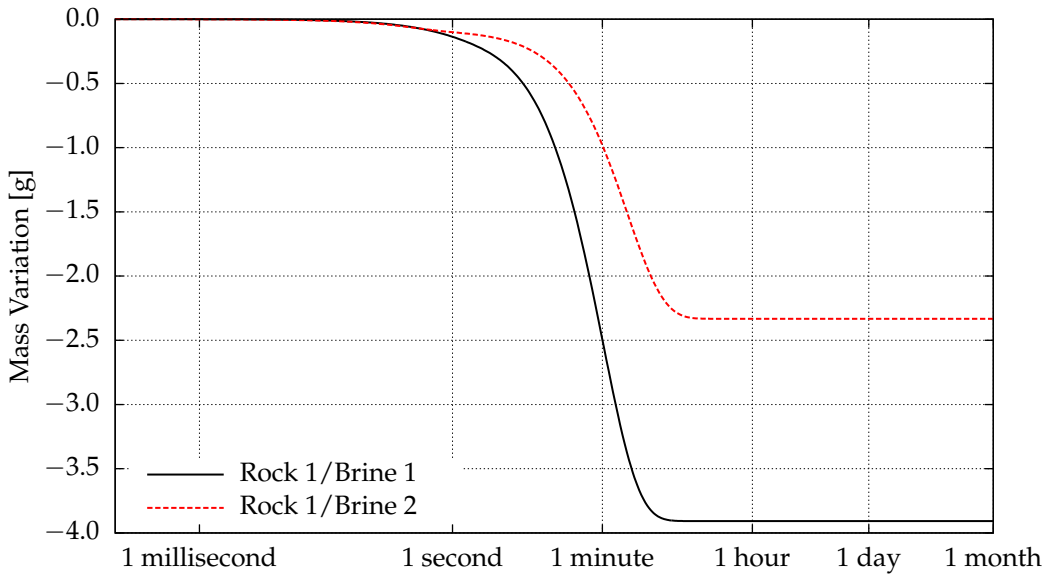


Figure 6.2: The transient state of mineral dolomite during one month of reaction with the CO_2 saturated subsurface fluid.

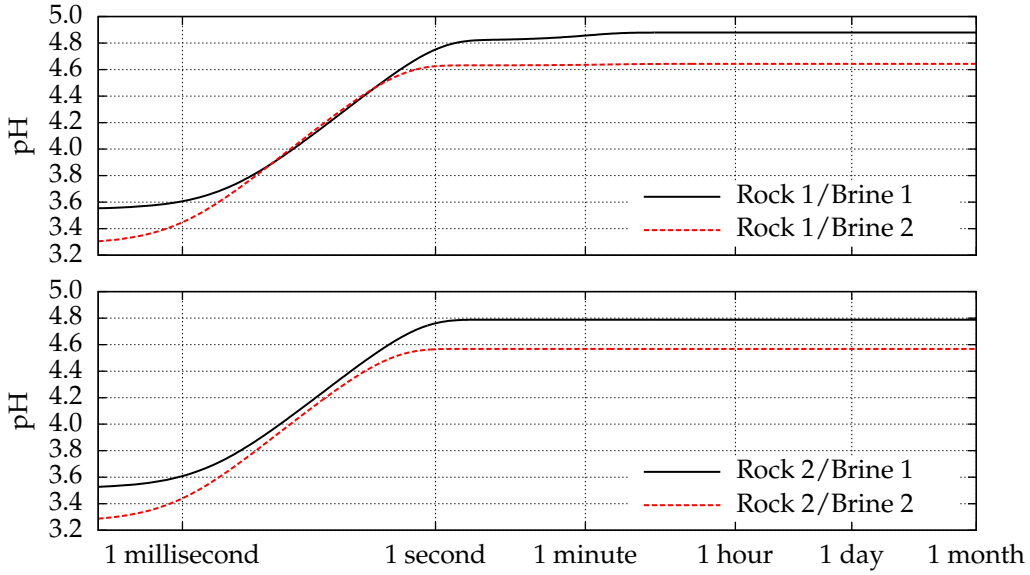


Figure 6.3: The transient state of the pH of the subsurface fluid during one month of reaction.

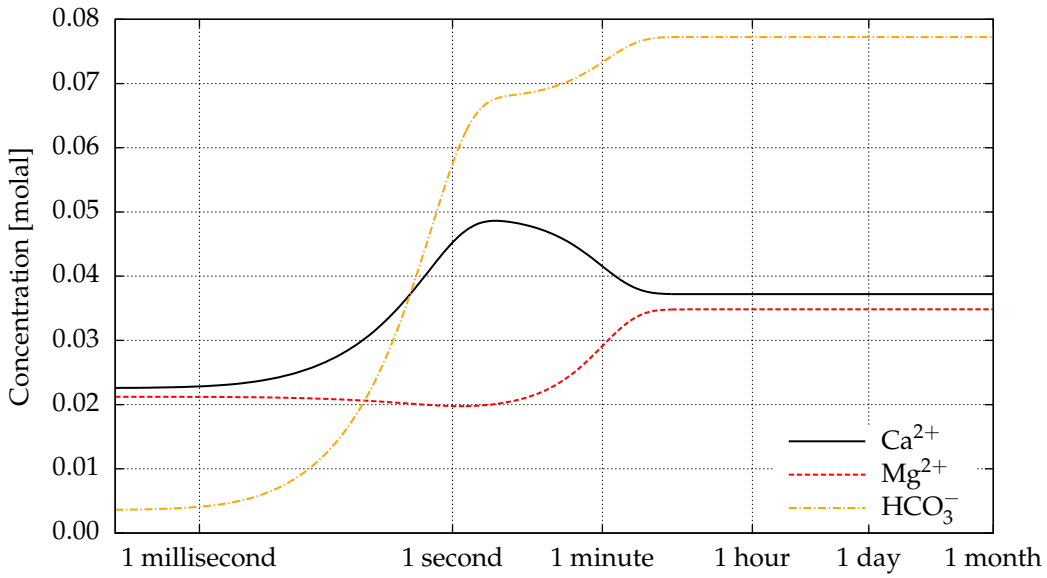


Figure 6.4: The transient state of ionic species Ca^{2+}, Mg^{2+} and HCO_3^- during one month of reaction.

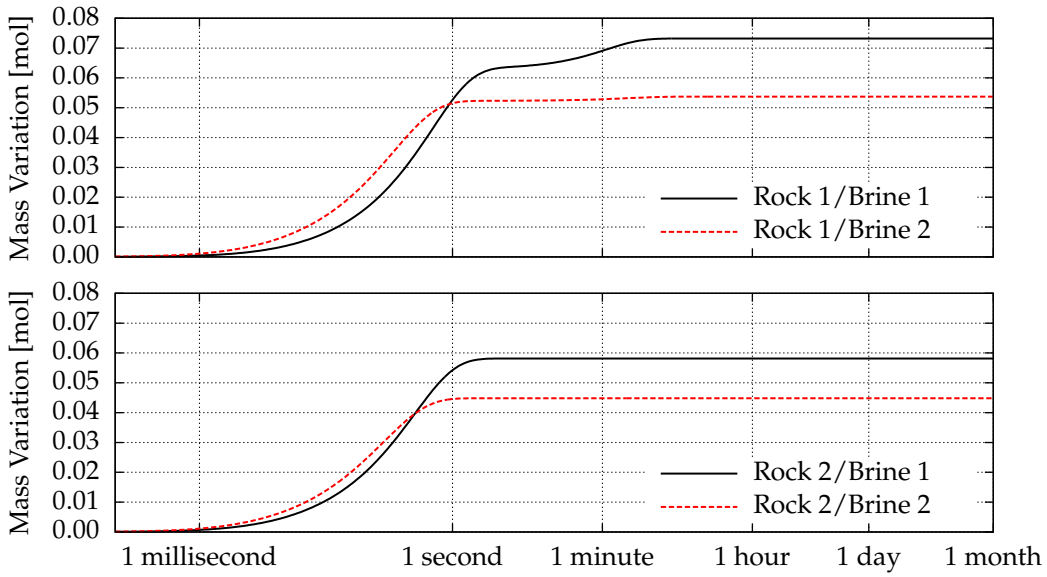


Figure 6.5: The transient increase in the amount of CO_2 in the subsurface fluid during one month of reaction.

Figure 6.5 presents the transient increase in the amount of CO_2 in the subsurface fluid. There are two carbon sources in the system from which this increase can occur: the carbonate minerals and the CO_2-rich phase. Since Figure 6.6 shows a simultaneous dissolution of supercritical CO_2, it follows that the dissolution of carbonate minerals increases the potential of brine to solubilise more carbon dioxide. This is an important water-gas-rock effect with considerable impact for safe carbon storage in saline aquifers, since in all cases the amount of supercritical CO_2 has decreased after one month of reactions.

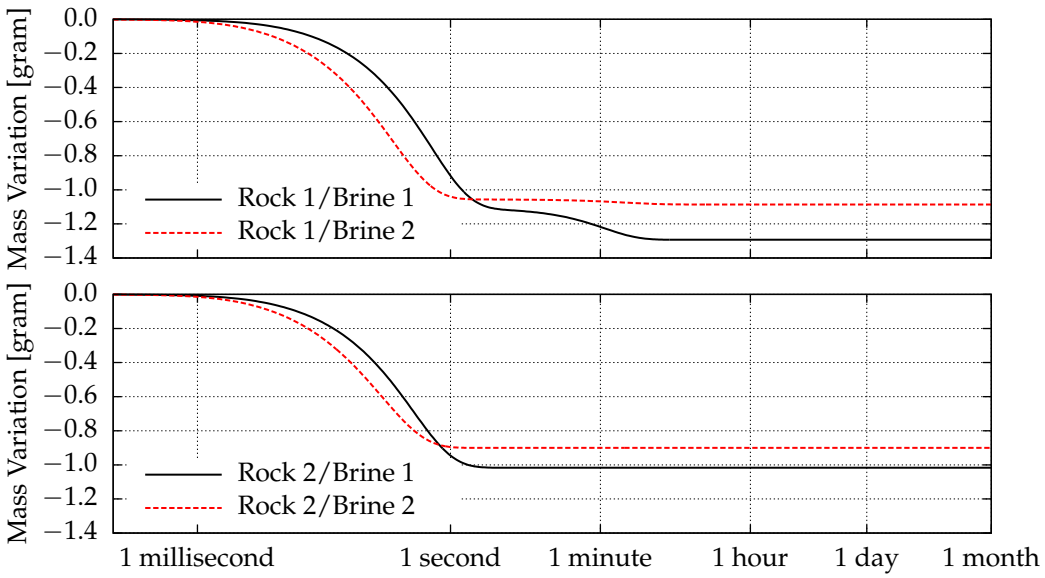


Figure 6.6: The transient state of supercritical CO_2 during one month of reaction.

6.2 Kinetic Modelling of CO_2 Injection into Sandstone Saline Aquifers

Consider now the kinetic modelling of the injection of supercritical CO_2 into a saline aquifer mainly composed of silicate minerals. The subsurface fluid is assumed to be the Qatari Brine 2 of the previous section, whose composition is shown in Table 6.2. The composition of sandstone rock was chosen based on a slightly modified composition of the Utsira formation in the Sleipner reservoir, North Sea, (Lagneau et al., 2005). The modification consisted of replacing a few original minerals by similar ones that could be found in the SUPCRT database (Johnson et al., 1992).

In addition, a small fraction of iron oxide (Fe_2O_3) was assumed in the rock. The idea was to investigate the precipitation of iron carbonates ($FeCO_3$) after the injection of CO_2. However, no formation of such carbonate minerals was observed. Nevertheless, the results shown next indicate that a fraction of the injected carbon dioxide is slowly transformed into calcium carbonates ($CaCO_3$) and then securely trapped for geological times.

Table 6.6: The composition of the sandstone rock before and after equilibrium with the subsurface fluid.

Mineral *(before equilibrium with brine)*	Volume (%)	Mineral *(after equilibrium with brine)*	Volume (%)
Quartz	77.0	Quartz	79.30
K-Feldspar	7.0	Muscovite	7.18
Calcite	6.0	Calcite	6.05
Muscovite	5.0	K-Feldspar	4.72
Albite	3.0	Hematite	2.01
Hematite	2.0	Phlogopite	0.75
Porosity:	10.0%	Porosity:	19.2%

Note: The volume of the rock sample is 300 cm^3.

The minerals composing the sandstone rock and their volume percent are listed in Table 6.6. To account for the long residency times of the subsurface fluid in the reservoir, the equilibrium state of the fluid and rock was calculated, resulting in a slightly different rock composition given on the right columns

Table 6.7: The chemical system representing the Qatari subsurface fluid and the sandstone rock.

Aqueous Phase					
Al^{3+}	ClO_3^-	$H_2(aq)$	$HS_2O_4^-$	$MgOH^+$	S_3^{2-}
AlO^+	ClO_4^-	$H_2O(l)$	$HSiO_3^-$	$MgSO_4(aq)$	$S_3O_6^{2-}$
AlO_2^-	$CO_2(aq)$	$H_2S(aq)$	HSO_3^-	Na^+	S_4^{2-}
$AlOH^{2+}$	CO_3^{2-}	$H_2S_2O_3(aq)$	HSO_4^-	$NaCl(aq)$	$S_4O_6^{2-}$
$Ca(HCO_3)^+$	Fe^{2+}	$H_2S_2O_4(aq)$	HSO_5^-	$NaHSiO_3(aq)$	S_5^{2-}
$Ca(HSiO_3)^+$	Fe^{3+}	$HAlO_2(aq)$	K^+	$NaOH(aq)$	$S_5O_6^{2-}$
Ca^{2+}	$FeCl^+$	$HCl(aq)$	$KCl(aq)$	$NaSO_4^-$	$SiO_2(aq)$
$CaCl^+$	$FeCl^{2+}$	$HClO(aq)$	$KHSO_4(aq)$	$O_2(aq)$	$SO_2(aq)$
$CaCl_2(aq)$	$FeCl_2(aq)$	$HClO_2(aq)$	$KOH(aq)$	OH^-	SO_3^{2-}
$CaCO_3(aq)$	$FeO(aq)$	HCO_3^-	KSO_4^-	S_2^{2-}	SO_4^{2-}
$CaOH^+$	FeO^+	$HFeO_2(aq)$	$Mg(HCO_3)^+$	$S_2O_3^{2-}$	$Sr(HCO_3)^+$
$CaSO_4(aq)$	FeO_2^-	$HFeO_2^-$	$Mg(HSiO_3)^+$	$S_2O_4^{2-}$	Sr^{2+}
Cl^-	$FeOH^+$	HO_2^-	Mg^{2+}	$S_2O_5^{2-}$	$SrCl^+$
ClO^-	$FeOH^{2+}$	HS^-	$MgCl^+$	$S_2O_6^{2-}$	$SrCO_3(aq)$
ClO_2^-	H^+	$HS_2O_3^-$	$MgCO_3(aq)$	$S_2O_8^{2-}$	$SrOH^+$
Gaseous Phase					
$H_2O(g)$	$CO_2(g)$	$O_2(g)$			
Mineral Phases					
Albite	Calcite	Forsterite	Kaolinite	Periclase	Siderite
Andradite	Chrysotile	Grossular	Lime	Phlogopite	Strontianite
Anhydrite	Cordierite	Halite	Magnesite	Prehnite	Tremolite
Annite	Diopside	Hematite	Magnetite	Pyrite	Wollastonite
Anorthite	Dolomite	Iron Oxide	Muscovite	Pyrophyllite	Zoisite
Brucite	Fayalite	K-Feldspar	Paragonite	Quartz	
Anthophyllite	Epidote	Jadeite	Nepheline	Pyrrhotite	

of the same table. This final rock composition was then used for the kinetic modelling of the injection of supercritical carbon dioxide into the system. Note that the sandstone rock is mainly composed of quartz and other silicate minerals, which are very unreactive. Therefore, simulation times in the order of millions of years were used to investigate the kinetics of the water-gas-rock effects following the CO_2 injection.

Similarly as before, the computational modelling requires the specification of the chemical system. This is given in Table 6.7, where the species composing the aqueous, gaseous and mineral phases of interest are presented. The chemical formulas of the minerals listed in this table are given in Table 6.8. Note that this system is substantially more complex than the previous one (see Table 6.3), used for the same injection modelling problem in a carbonate aquifer. For example, there are now 40 pure mineral phases, while the simulation with the carbonate rock only required 5 minerals.

Table 6.9 shows the kinetic species and the reactions assumed to be controlled by kinetics in the calculations. Note that even though the reaction rates of carbonate minerals are substantially faster than silicate minerals, they were also assumed to be controlled by kinetics. However, the carbonate minerals could have been assumed to be controlled by equilibrium without sacrificing much the accuracy of the calculations. We observed that this would only anticipate some eventual precipitation of carbonate minerals by just a few years, which is an insignificant time

Table 6.8: The list of mineral species composing the sandstone rock and their respective chemical formula.

Mineral	Chemical Formula	Mineral	Chemical Formula
Albite	$Na(AlSi_3)O_8$	Fayalite	Fe_2SiO_4
Annite	$KFe_3(AlSi_3)O_{10}(OH)_2$	Epidote	$Ca_2FeAl_2Si_3O_{12}(OH)$
Anorthite	$Ca(Al_2Si_2)O_8$	Andradite	$Ca_3Fe_2Si_3O_{12}$
Calcite	$CaCO_3$	Anhydrite	$CaSO_4$
Dolomite	$CaMg(CO_3)_2$	Anthophyllite	$Mg_7Si_8O_{22}(OH)_2$
Iron Oxide	FeO	Brucite	$Mg(OH)_2$
Halite	$NaCl$	Chrysotile	$Mg_3Si_2O_5(OH)_4$
Hematite	Fe_2O_3	Cordierite	$Mg_2Al_3(AlSi_5)O_{18}$
K-Feldspar	$K(AlSi_3)O_8$	Diopside	$CaMg(SiO_3)_2$
Kaolinite	$Al_2Si_2O_5(OH)_4$	Forsterite	Mg_2SiO_4
Lime	CaO	Grossular	$Ca_3Al_2Si_3O_{12}$
Magnesite	$MgCO_3$	Jadeite	$NaAl(SiO_3)_2$
Muscovite	$KAl_2(AlSi_3)O_{10}(OH)_2$	Nepheline	$Na(AlSi)O_4$
Periclase	MgO	Paragonite	$NaAl_2(AlSi_3)O_{10}(OH)_2$
Quartz	SiO_2	Phlogopite	$KMg_3(AlSi_3)O_{10}(OH)_2$
Siderite	$FeCO_3$	Prehnite	$Ca_2Al_2Si_3O_{10}(OH)_2$
Strontianite	$SrCO_3$	Pyrophyllite	$Al_2Si_4O_{10}(OH)_2$
Magnetite	Fe_3O_4	Tremolite	$(Ca_2Mg_5)Si_8O_{22}(OH)_2$
Pyrite	FeS_2	Wollastonite	$CaSiO_3$
Pyrrhotite	FeS	Zoisite	$Ca_2Al_3Si_3O_{12}(OH)$

Table 6.9: The kinetic species and reactions for the kinetic modelling in a sandstone saline aquifer.

Kinetic Species	**Reactions**
Calcite	$\text{Calcite} + H^+ \rightleftharpoons HCO_3^- + Ca^{2+}$
Magnesite	$\text{Magnesite} + H^+ \rightleftharpoons HCO_3^- + Mg^{2+}$
Siderite	$\text{Siderite} + H^+ \rightleftharpoons HCO_3^- + Fe^{2+}$
Strontianite	$\text{Strontianite} + H^+ \rightleftharpoons HCO_3^- + Sr^{2+}$
Dolomite	$\text{Dolomite} + 2H^+ \rightleftharpoons 2HCO_3^- + Ca^{2+} + Mg^{2+}$
Quartz	$\text{Quartz} \rightleftharpoons SiO_2(aq)$
Kaolinite	$\text{Kaolinite} + 6H^+ \rightleftharpoons 2Al^{3+} + 5H_2O(l) + 2SiO_2(aq)$
Albite	$\text{Albite} + 4H^+ \rightleftharpoons Al^{3+} + Na^+ + 2H_2O(l) + 3SiO_2(aq)$
K-Feldspar	$\text{K-Feldspar} + 4H^+ \rightleftharpoons Al^{3+} + K^+ + 2H_2O(l) + 3SiO_2(aq)$
Anorthite	$\text{Anorthite} + 8H^+ \rightleftharpoons 2Al^{3+} + Ca^{2+} + 4H_2O(l) + 2SiO_2(aq)$
Muscovite	$\text{Muscovite} + 10H^+ \rightleftharpoons 3Al^{3+} + K^+ + 6H_2O(l) + 3SiO_2(aq)$
Annite	$\text{Annite} + 10H^+ \rightleftharpoons Al^{3+} + K^+ + 3Fe^{2+} + 6H_2O(l) + 3SiO_2(aq)$
Hematite	$\text{Hematite} + 6H^+ \rightleftharpoons 2Fe^{3+} + 3H_2O(l)$

scale when compared to millions of years.

Equation (4.42) was used for the calculation of the mineral rates of dissolution and precipitation. The rate parameters of each mineral in Table 6.9 were taken from Palandri and Kharaka (2004). The rate parameters for the carbonate minerals siderite and strontianite were assumed the same as those for calcite. The specific surface area of the kinetic minerals were assumed to be $5000\,m^2/m^3$ for the whole calculation.

Figure 6.7 shows the kinetic dissolution and precipitation of the silicate minerals over a period of one hundred million years. The initial time corresponds to the moment the injected super-

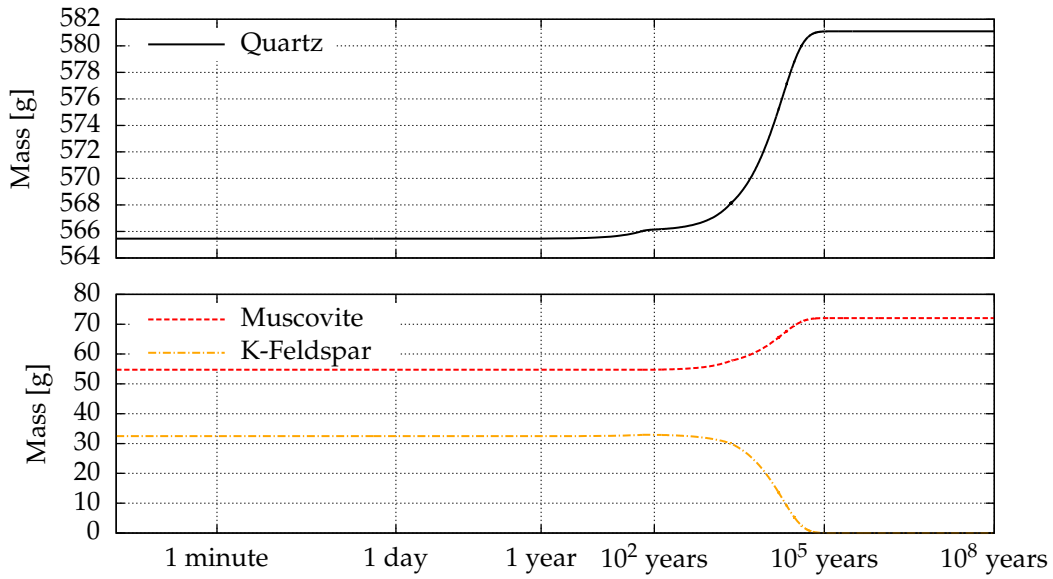


Figure 6.7: The kinetic dissolution and precipitation of the major silicate minerals in the sandstone rock during one hundred million years.

critical CO_2 has equilibrated with the subsurface fluid, which was assumed to be considerably faster than any interaction between brine and the kinetic minerals. A constant temperature and pressure of 60 °C and 150 bar was assumed throughout the calculation. Note that it takes about a few thousand years for the start of some significant changes in the silicate composition of the sandstone rock. It can be observed that quartz and muscovite precipitates afterwards, competing for the silica produced from the complete dissolution of K-feldspar. Any absent silicate mineral in Figure 6.7 remained completely dissolved during the simulation.

Figure 6.8 shows the dissolution and precipitation behaviour of calcite over millions of years. Any other carbonate mineral remained fully dissolved in the brine for the duration of the simulation. Overall, this figure shows that a significant amount of precipitation of calcite follows after the injection of supercritical carbon dioxide, possibly sequestering the injected greenhouse gas. This happens in two stages: one right after the injection of CO_2, which is consistent with the high reactivity of calcite, and another after a few thousands years, when the silicate minerals start to slowly dissolve and precipitate.

The previous result alone is not conclusive to determine the extent at which the injected carbon dioxide can be trapped as precipitated carbonate minerals. To determine how much of the injected carbon dioxide is trapped as precipitated carbonate minerals, the amount of carbon in the aqueous, gaseous and mineral phases over one hundred million years is shown in Figure 6.9. This figure shows that a small amount of mobile carbon dioxide is produced after one day of reaction until about 2300 years. Afterwards, supercritical CO_2 is continuously consumed until about 70000 years, where nearly 0.83 g of the greenhouse gas had been converted into carbonate minerals. The same figure also shows that 1.66 g of dissolved carbon dioxide in the brine is transformed into precipitated carbonate minerals.

The kinetic modelling assumed an injection of 440 g of supercritical carbon dioxide into the system, from which about 36.0 g was instantaneously dissolved into the brine before any inter-

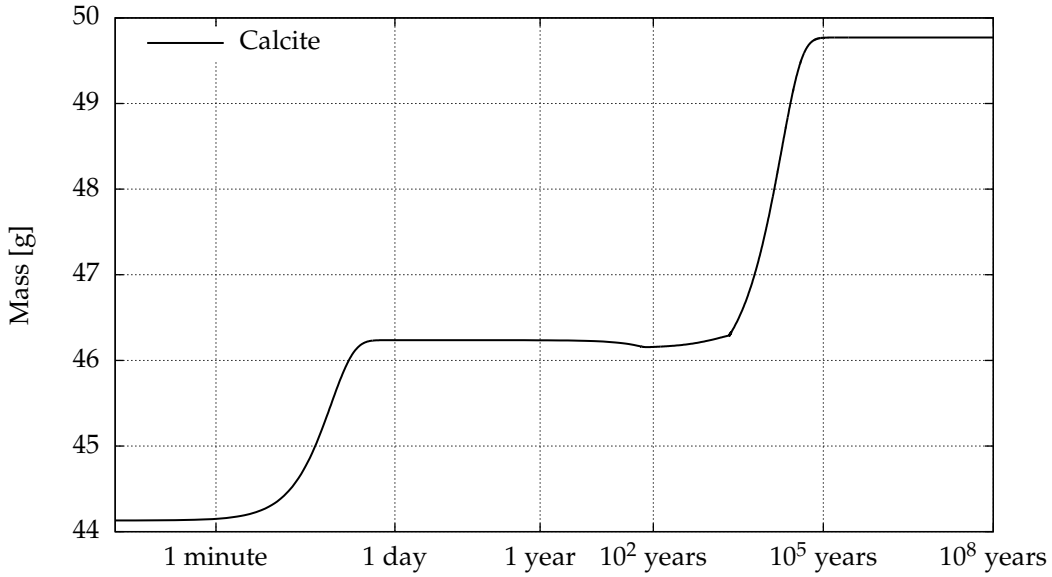


Figure 6.8: The kinetic dissolution and precipitation of calcite in the sandstone rock during one hundred million years.

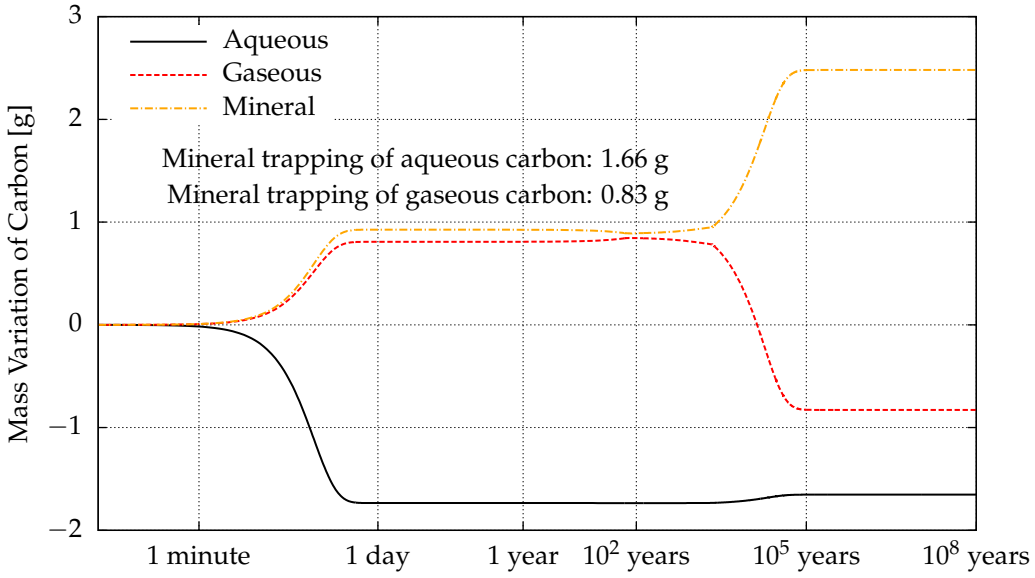


Figure 6.9: The kinetic trapping of mobile carbon (aqueous and gaseous/supercritical) as precipitated carbonate minerals during one hundred million years.

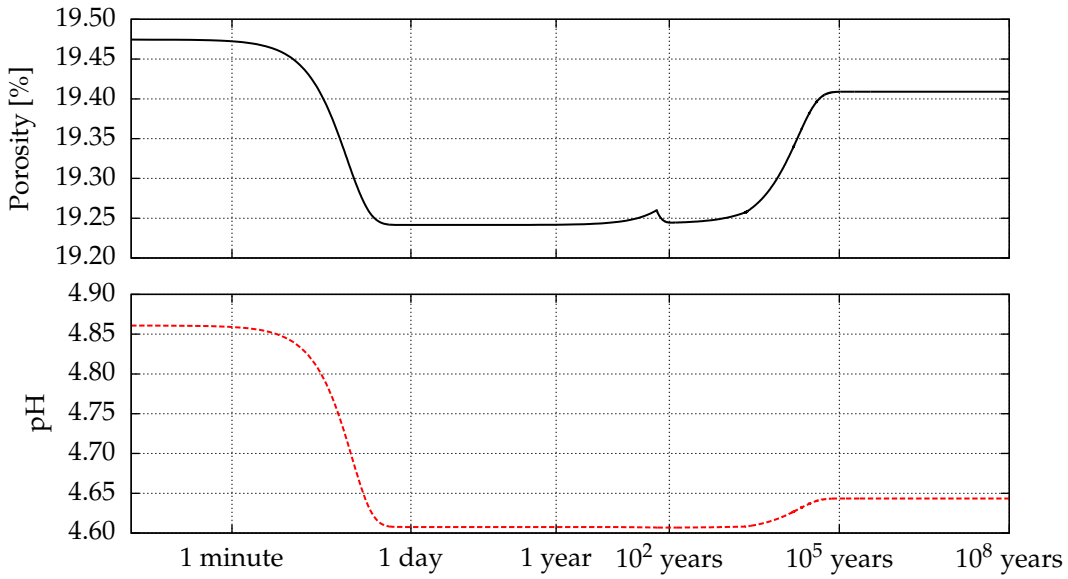


Figure 6.10: The kinetic evolution of the porosity of the sandstone rock and acidity of the subsurface fluid in reservoir during one hundred million years.

action between fluid and rock was allowed to take place. Thus we see that in the long term, the amount of trapped carbon dioxide (by solubility and mineral mechanisms) in a saline aquifer is increased by nearly 7% due to the precipitation of carbonate minerals. This represents a significant increase in the safe storage capability of sandstone aquifers. It should be remarked, however, that this increase only takes place in the order of tens of thousands of years.

Figure 6.10 shows how the porosity of the sandstone rock and the pH of the subsurface fluid vary over millions of years after CO_2 injection. Comparing the porosity curve with the calcite curve of Figure 6.8, it can be observed that porosity decreases due to the precipitation of calcite during the first few hours. The porosity increases, after thousands of years, when K-feldspar starts to dissolve away, as shown in Figure 6.7, even though calcite is simultaneously precipitating. This happens because a larger amount of K-feldspar is dissolved in the process. The effect of pH on the porosity of this sandstone rock is not as expressive as it is in carbonate rocks, where there can be significant dissolution and formation of wormholes (Gaus et al., 2008). Here, for example, a decrease in pH (or increase in acidity) is not accompanied by an increase in porosity, as it can be seen after one day of reaction in Figure 6.10.

6.3 Kinetic Modelling of Pure Calcite Dissolution

This section presents a kinetic modelling of the calcite dissolution problem in CO_2 saturated water.

Peng et al. (2014) recently determined the dissolution rates of calcite in CO_2 saturated waters far from equilibrium. The following major parallel reactions were taken into account in their

study:

$$\mathrm{CaCO_3(s) + H^+ \xrightarrow{k_1} Ca^{2+} + HCO_3^-}, \tag{6.1}$$

$$\mathrm{CaCO_3(s) + H_2CO_3^\star \xrightarrow{k_2} Ca^{2+} + 2HCO_3^-}, \tag{6.2}$$

where $\mathrm{H_2CO_3}^*$ represents a sum of dissolved molecular species $\mathrm{CO_2(aq)}$ and $\mathrm{H_2CO_3(aq)}$, and k_1 and k_2 are the rate constants of the reactions. These rate constants are used to compute the rates of their respective reactions as:

$$r_1 = k_1 a_{\mathrm{H^+}} \tag{6.3}$$

and

$$r_2 = k_2 a_{\mathrm{CO_2(aq)}}, \tag{6.4}$$

where $a_{\mathrm{H^+}}$ and $a_{\mathrm{CO_2(aq)}}$ are the activities of the species $\mathrm{H^+}$ and $\mathrm{CO_2(aq)}$ respectively.

It is not very clear how the artificial species $\mathrm{H_2CO_3}^*$ can be represented computationally. Because of this, reaction (6.2) has been replaced in this work by:

$$\mathrm{CaCO_3(s) + H_2O(l) + CO_2(aq) \xrightarrow{k_2} Ca^{2+} + 2HCO_3^-}. \tag{6.5}$$

The reaction rates given by equations (6.3) and (6.4) were determined far from equilibrium. Near equilibrium, these rates should continuously decrease to zero in order to reproduce the eventual equilibrium state of the mixture of calcite and $\mathrm{CO_2}$ saturated water. To account for this behaviour, the following rate equations were adopted instead:

$$r_1 = k_1 a_{\mathrm{H^+}} \left(1 - \frac{Q_1}{K_1}\right) \tag{6.6}$$

and

$$r_2 = k_2 a_{\mathrm{CO_2(aq)}} \left(1 - \frac{Q_2}{K_2}\right), \tag{6.7}$$

where K_1 and K_2 are the equilibrium constants of reactions (6.1) and (6.5); and Q_1 and Q_2 are the reaction quotients of the same reactions, given by:

$$Q_1 = \frac{a_{\mathrm{Ca^{2+}}} a_{\mathrm{HCO_3^-}}}{a_{\mathrm{CaCO_3(s)}} a_{\mathrm{H^+}}} \tag{6.8}$$

and

$$Q_2 = \frac{a_{\mathrm{Ca^{2+}}} a^2_{\mathrm{HCO_3^-}}}{a_{\mathrm{CaCO_3(s)}} a_{\mathrm{H_2O(l)}} a_{\mathrm{CO_2(aq)}}}. \tag{6.9}$$

These introduced factors have negligible effect when calcite is far from saturation, that is, when $0 < Q_i/K_i \ll 1$. At equilibrium, calcite is saturated and $Q_i/K_i = 1$ for both reactions.

Figure 6.11 presents the calculated concentration of calcium in the $\mathrm{CO_2}$ saturated water while calcite continuously dissolves. These calculations were compared with the experimental measurements of Peng et al. (2014) at $T = 353\,\mathrm{K}$ and $P = 13.8\,\mathrm{MPa}$. It can be seen that a good accuracy is achieved for the period of time between 5 minutes and 1 hour for which experimental measurements were available.

The calculations indicate that 30 hours would be necessary for the system to establish equilibrium. However, this equilibration time is possibly inaccurate due to several uncertainties in

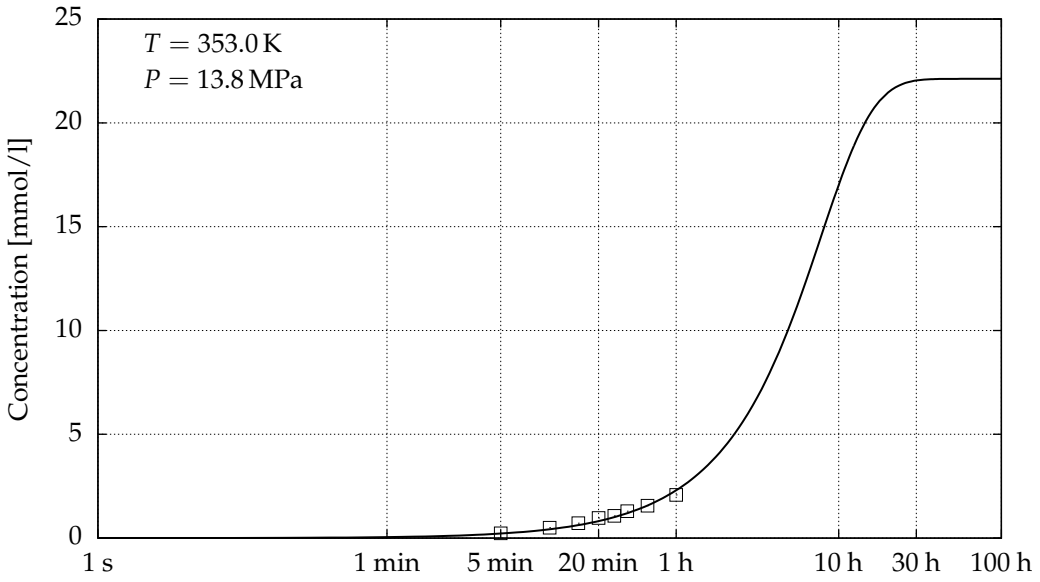


Figure 6.11: Comparison of the calculated concentration of calcium in the aqueous solution (line) with the experimental data (points) of Peng et al. (2014) during dissolution of calcite in a CO_2 saturated water.

the computational modelling. For example, the specific surface area of the mineral remained constant at 208 mm^2 for the entire simulation. In addition, the introduced saturation factors in the reaction rates (6.6) and (6.7) are not necessarily correct, and some adjustments such as:

$$r_1 = k_1 a_{H^+} \left[1 - \left(\frac{Q_1}{K_1} \right)^{\xi_1} \right]^{\eta_1} \tag{6.10}$$

and

$$r_2 = k_2 a_{CO_2(aq)} \left[1 - \left(\frac{Q_2}{K_2} \right)^{\xi_2} \right]^{\eta_2}, \tag{6.11}$$

might be necessary to improve the accuracy of the rates, where the exponents ξ_i, η_i would be determined experimentally.

An alternative to the use of equations (6.10) and (6.11) to model the rates near equilibrium, and also when the mineral is precipitating, would be a combined use of the forward and reverse rates of the mineral reaction. The difficulty lies, however, on determining accurate models for precipitation rates, since it depends on several factors such as surface area, nucleation and crystal growth mechanisms.

CHAPTER 7

Conclusions and Future Work

Computational methods are essential for geochemical modelling. Analysing even the simplest geochemical reactions that occur in nature can be a daunting task without the use of computational algorithms. Most of the environmental problems occurring at the Earth's surface, subsurface and mantle can be tackled by direct or indirect chemical equilibrium and kinetics calculations. This work has proposed novel numerical methods for these computations.

It suffices to assume chemical equilibrium in many environmental problems. These calculations can be formulated by either a stoichiometric or non-stoichiometric approach. The former solves a system of non-linear mass-action and mass-balance equations, while the later solves a constrained optimisation problem, where the Gibbs free energy of the system is minimised. This work has proposed equilibrium algorithms based on both approaches, as described in Chapters 2 and 3. The Gibbs energy minimisation approach is based on a primal-dual interior-point method that has been demonstrated to be very efficient and robust. Its analysed performance suggested its potential integration into reactive transport simulators.

Geochemical reactions proceed with different speeds: some with time scales in the order of years, and others in the order of microseconds. Thus, chemical kinetic modelling of geochemical processes very often assumes partial equilibrium in the system, where some fast reactions are selected to be governed by thermodynamic equilibrium. Chapter 4 proposed an innovative numerical method for chemical kinetic modelling of geochemical systems. It adopts an implicit scheme of integration of the algebraic-differential equations in order to achieve robustness and efficiency. The method uses the Gibbs energy minimisation method to account for the geochemical reactions governed by thermodynamic equilibrium.

Geochemical modelling can take advantage of the rich set of thermodynamic models available in the literature. By using a variety of such models for aqueous, gaseous and mineral species, this work demonstrated in Chapter 5 the capability to solve multiphase equilibrium problems relevant to carbon dioxide sequestration in saline aquifers accurately. Most of these calculations have been assessed against recent experimental measurements of carbon dioxide solubility in NaCl and non-NaCl brines. Comparison of pH calculations with experimental data was also done, as well as benchmarking some calculations against other geochemical solvers.

Complicated water-gas-rock effects take place during injection of carbon dioxide into saline aquifers. Understanding these interactions and their long term effects is fundamental for safety assessment of carbon capture and storage. As shown in Chapter 6, taking into account the interactions among the subsurface fluid, supercritical carbon dioxide and the rock-forming minerals indicated that carbonate rocks increase the potential of brines to dissolve even more mobile supercritical CO_2 and sandstone rocks have the potential to store substantial amounts of CO_2 as precipitated carbonate minerals in the long term.

As a future work, it remains to integrate these algorithms into a reactive transport simulator. This will allow a more comprehensive understanding of the fate of injected carbon dioxide into saline aquifers, where both transport and chemical processes are taken into account.

Development of such simulator is already in progress. For this, the FEniCS Project (Logg and Wells, 2010; Logg et al., 2011) has been adopted, which consists of a collection of tools for solving partial differential equations using finite element methods in one-, two- or three-dimensional unstructured meshes. This choice has proved to be very efficient in developing complicated codes for flow simulations, since most of the difficulties in handling meshes, assembling vectors and matrices from the elements, performing mesh adaptivity and parallelism, and so forth has been abstracted away from the user.

In addition, further investigation of other optimisation methods are needed for chemical equilibrium calculations. The use of interior-point methods, as seen before, has resulted in solutions where phases that should be inexistent at equilibrium are in fact present at low amounts. These low amounts are in the same order of magnitude of the perturbation parameter used for the interior-point minimisation algorithm. Thus, in the end of the calculation, it is always necessary to apply a refinement calculation to correct the distribution of mass among the stable phases (see Section 3.3.8).

Another class of methods, called active-set methods (Nocedal and Wright, 1999; Fletcher, 2000), should be tested for chemical equilibrium calculations. In these methods, there is no need to use a perturbation parameter as in the interior-point method, and the species that should be inexistent will have an exact zero amount. As a result, this approach would no longer require a phase stability test as shown in Section 3.3.8, since this comes as a direct result. It is expected, therefore, that some performance gain can be obtained with active-set methods.

These chemical equilibrium and kinetics methods have been implemented in REAKTORO, a scientific library, written in the C++ programming language, for computational geochemical modelling. The code is freely available at `www.bitbucket.org/reaktoro` and released under the open-source GPL license.

BIBLIOGRAPHY

Aagaard, P. and Helgeson, H. C. (1982). Thermodynamic and kinetic constraints on reaction rates among minerals and aqueous solutions: I. Theoretical considerations. *American Journal of Science*, 282(3):237–285.

Allison, J. and Kevin, J. (1991). MINTEQA2/PRODEFA2: A Geochemical Assessment Model for Environmental Systems: Version 3.0 User's Manual. Technical Report 600/3-91/021.

Anderson, G. M. and Crerar, D. A. (1993). *Thermodynamics in Geochemistry: The Equilibrium Model*. Oxford University Press, New York.

André, L., Audigane, P., Azaroual, M., and Menjoz, A. (2007). Numerical modeling of fluid-rock chemical interactions at the supercritical CO2-liquid interface during CO2 injection into a carbonate reservoir, the Dogger aquifer (Paris Basin, France). *Energy conversion and management*, 48(6):1782–1797.

Ascher, U. M. and Petzold, L. R. (1998). *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. Society for Industrial Mathematics. ISBN 0898714125.

Audigane, P., Gaus, I., Czernichowski-Lauriol, I., Pruess, K., and Xu, T. (2007). Two-dimensional reactive transport modeling of CO2 injection in a saline aquifer at the Sleipner site, North Sea. *American Journal of Science*, 307(7):974–1008.

Bale, C., Bélisle, E., Chartrand, P., Decterov, S., Eriksson, G., Hack, K., Jung, I.-H., Kang, Y.-B., Melançon, J., Pelton, A., Robelin, C., and Petersen, S. (2009). FactSage thermochemical software and databases—recent developments. *Calphad*, 33(2):295–311.

Bale, C., Chartrand, P., Degterov, S., Eriksson, G., Hack, K., Ben Mahfoud, R., Melançon, J., Pelton, A., and Petersen, S. (2002). FactSage thermochemical software and databases. *Calphad*, 26(2):189–228.

Ball, J., Nordstrom, D., and Zachmann, D. (1987). WATEQ4F–A personal computer FORTRAN translation of the geochemical model WATEQ2 with revised data base. *U.S. Geological Survey Open-File Report*, 87(50):108.

Benson, H., Shanno, D. F., and Vanderbei, R. J. (2000). Interior-point methods for nonconvex nonlinear programming: Jamming and comparative numerical testing. *Operations Research and Financial Engineering, Princeton University, ORFE-00-02*.

Bethke, C. M. (2007). *Geochemical and Biogeochemical Reaction Modeling*. Cambridge University Press, New York, 2 edition.

Brown, P. N., Byrne, G. D., and Hindmarsh, A. C. (1989). VODE: A Variable-Coefficient ODE Solver. *SIAM Journal on Scientific and Statistical Computing*, 10(5):1038–1051.

Byrd, R. H., Hribar, M. E., and Nocedal, J. (1999). An Interior Point Algorithm for Large-Scale Nonlinear Programming. *SIAM Journal on Optimization*, 9(4):877–900.

Chamberlain, R., Powell, M., Lemarechal, C., and Pedersen, H. (1982). The watchdog technique for forcing convergence in algorithms for constrained optimization. *Mathematical Programming Studies*, 16:1–17.

Cohen, S. D. and Hindmarsh, A. C. (1996). CVODE, a stiff/nonstiff ODE solver in C. *Computers in Physics*, 10(2):138–143.

Costa, M. F. P. and Fernandes, E. M. (2011a). Assessing the potential of interior point barrier filter line search methods: nonmonotone versus monotone approach. *Optimization*, 60(10-11):1251–1268.

Costa, M. F. P. and Fernandes, E. M. G. P. (2008). Practical implementation of an interior point nonmonotone line search filter method. *International Journal of Computer Mathematics*, 85(3-4):397–409.

Costa, M. F. P. and Fernandes, E. M. G. P. (2011b). On Minimizing Objective and KKT Error in a Filter Line Search Strategy for an Interior Point Method. In Murgante, B., Gervasi, O., Iglesias, A., Taniar, D., and Apduhan, B., editors, *Computational Science and Its Applications - ICCSA 2011*, volume 6784 of *Lecture Notes in Computer Science*, pages 231–244. Springer Berlin Heidelberg.

de Capitani, C. and Brown, T. H. (1987). The computation of chemical equilibrium in complex systems containing non-ideal solutions. *Geochimica et Cosmochimica Acta*, 51(10):2639–2652.

Drummond, S. (1981). *Boiling and mixing of hydrothermal fluids: chemical effects on mineral precipitation*. Ph.d., Pennsylvania State University.

Duan, Z., Møller, N., and Weare, J. H. (1992). An equation of state for the CH4–CO2–H2O system: I. Pure systems from 0 to 1000°C and 0 to 8000 bar. *Geochimica et Cosmochimica Acta*, 56(7):2605–2617.

Duan, Z. and Sun, R. (2003). An improved model calculating CO2 solubility in pure water and aqueous NaCl solutions from 273 to 533 K and from 0 to 2000 bar. *Chemical Geology*, 193(3-4):257–271.

Duan, Z., Sun, R., Zhu, C., and Chou, I.-M. (2006). An improved model for the calculation of CO2 solubility in aqueous solutions containing Na+, K+, Ca2+, Mg2+, Cl-, and SO42-. *Marine Chemistry*, 98(2-4):131–139.

El-Bakry, A. S., Tapia, R. A., Tsuchiya, T., and Zhang, Y. (1996). On the formulation and theory of the Newton interior-point method for nonlinear programming. *Journal of Optimization Theory and Applications*, 89(3):507–541.

El-Maghraby, R. M. and Blunt, M. J. (2013). Residual CO2 trapping in Indiana limestone. *Environmental science & technology*, 47(1):227–33.

Ennis-King, J. and Paterson, L. (2005). Role of Convective Mixing in the Long-Term Storage of Carbon Dioxide in Deep Saline Formations. *SPE Journal*, 10(3):5–8.

Ennis-King, J. and Paterson, L. (2007). Coupling of geochemical reactions and convective mixing in the long-term geological storage of carbon dioxide. *International Journal of Greenhouse Gas Control*, 1(1):86–93.

Eriksson, G. and Hack, K. (1990). ChemSage—A computer program for the calculation of complex chemical equilibria. *Metallurgical Transactions B*, 21(6):1013–1023.

Fiacco, A. V. and McCormick, G. P. (1990). *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*. Society for Industrial and Applied Mathematics.

Fletcher, R. (2000). *Practical Methods of Optimization*. Wiley, second edition.

Fletcher, R. and Leyffer, S. (2002). Nonlinear programming without a penalty function. *Mathematical Programming*, 91(2):239–269.

Garcia, M., Dávila, G., Offeddu, F., Soler, J. M., and Cama, J. (2011). Reactions During CO2 Geological Sequestration: Dissolution of Calcite and Dolomite Coupled to Gypsum Precipitation. *Revista de la Sociedad Española de Mineralogía*, pages 93–94.

Garrels, R. M. and Thompson, M. E. (1962). A chemical model for sea water at 25 degrees C and one atmosphere total pressure. *American Journal of Science*, 260(1):57–66.

Gaus, I., Audigane, P., Andre, L., Lions, J., Jacquemet, N., Durst, P., Czernichowski-Lauriol, I., and Azaroual, M. (2008). Geochemical and solute transport modelling for CO2 storage, what to expect from it? *International Journal of Greenhouse Gas Control*, 2(4):605–625.

Gill, P. E., Murray, W., and Saunders, M. a. (2002). SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization. *SIAM Journal on Optimization*, 12(4):979–1006.

Gill, P. E., Murray, W., and Saunders, M. A. (2005). Users Guide for SQOPT Version 7: A Fortran Package for Large-Scale Linear and Quadratic Programming. *Numerical Analysis Report NA 97-4*.

Gunter, W., Pratt, A., Buschkuehle, B., and Perkins, E. (2004). Acid gas injection in the Brazeau Nisku Q carbonate reservoir: Geochemical reactions as a result of the injection of an H2S-CO2 waste stream. In *Proceedings of the 7th International Conference on Greenhouse Gas Control Technologies*, volume 1, pages 5–9.

Gunter, W. D., Perkins, E. H., and McCann, T. J. (1993). Aquifer disposal of CO2-rich gases: Reaction design for added capacity. *Energy Conversion and Management*, 34(9-11):941–948.

Gunter, W. D., Wiwehar, B., and Perkins, E. H. (1997). Aquifer disposal of CO2-rich greenhouse gases: Extension of the time scale of experiment for CO2-sequestering reactions by geochemical modelling. *Mineralogy and Petrology*, 59(1-2):121–140.

Hairer, E., Nørsett, S. P., and Wanner, G. (2008). *Solving Ordinary Differential Equations I: Nonstiff Problems*, volume 1. Springer, 2nd edition. ISBN 3540566708.

Hairer, E. and Wanner, G. (2010). *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*, volume 2. Springer, 2nd edition. ISBN 3642052207.

Harvey, J.-P., Eriksson, G., Orban, D., and Chartrand, P. (2013). Global minimization of the Gibbs energy of multicomponent systems involving the presence of order/disorder phase transitions. *American Journal of Science*, 313(3):199–241.

Harvie, C. E., Greenberg, J. P., and Weare, J. H. (1987). A chemical equilibrium algorithm for highly non-ideal multiphase systems: Free energy minimization. *Geochimica et Cosmochimica Acta*, 51(5):1045–1057.

Harvie, C. E., Mø ller, N., and Weare, J. H. (1984). The prediction of mineral solubilities in natural waters: The Na-K-Mg-Ca-H-Cl-SO4-OH-HCO3-CO3-CO2-H2O system to high ionic strengths at 25°C. *Geochimica et Cosmochimica Acta*, 48(4):723–751.

Helgeson, H. (1968). Evaluation of irreversible reactions in geochemical processes involving minerals and aqueous solutions—I. Thermodynamic relations. *Geochimica et Cosmochimica Acta*, 32(8):853–877.

Helgeson, H. (1971). Kinetics of mass transfer among silicates and aqueous solutions. *Geochimica et Cosmochimica Acta*, 35(5):421–469.

Helgeson, H., Brown, T., Nigrini, A., and Jones, T. (1970). Calculation of mass transfer in geochemical processes involving aqueous solutions. *Geochimica et Cosmochimica Acta*, 34(5):569–592.

Helgeson, H., Delany, J., Nesbitt, H., and Bird, D. (1978). Summary and critique of the thermodynamic properties of rock-forming minerals. *American Journal of Science*, 278 A(1).

Helgeson, H. C. (1969). Thermodynamics of hydrothermal systems at elevated temperatures and pressures. *American Journal of Science*, 267(7):729–804.

Helgeson, H. C., Garrels, R. M., and MacKenzie, F. T. (1969). Evaluation of irreversible reactions in geochemical processes involving minerals and aqueous solutions—II. Applications. *Geochimica et Cosmochimica Acta*, 33(4):455–481.

Helgeson, H. C. and Kirkham, D. H. (1974a). Theoretical prediction of the thermodynamic behavior of aqueous electrolytes at high pressures and temperatures: I. Summary of the thermodynamic/electrostatic properties of the solvent. *American Journal of Science*, 274(10):1089–1198.

Helgeson, H. C. and Kirkham, D. H. (1974b). Theoretical prediction of the thermodynamic behavior of aqueous electrolytes at high pressures and temperatures: II. Debye-Huckel parameters for activity coefficients and relative partial molal properties. *American Journal of Science*, 274(10):1199–1261.

Helgeson, H. C. and Kirkham, D. H. (1976). Theoretical prediction of the thermodynamic properties of aqueous electrolytes at high pressures and temperatures: III. Equation of state for aqueous species at infinite dilution. *American Journal of Science*, 276(2):97–240.

Helgeson, H. C., Kirkham, D. H., and Flowers, G. C. (1981). Theoretical prediction of the thermodynamic behavior of aqueous electrolytes at high pressures and temperatures: IV. Calculation of activity coefficients, osmotic coefficients, and apparent molal and standard and relative partial molal properties to 600 C. *American Journal of Science*, 281(10):1249–1516.

Helgeson, H. C. and Murphy, W. M. (1983). Calculation of mass transfer among minerals and aqueous solutions as a function of time and surface area in geochemical processes. I. computational approach. *Journal of the International Association for Mathematical Geology*, 15(1):109–130.

Helgeson, H. C., Murphy, W. M., and Aagaard, P. (1984). Thermodynamic and kinetic constraints on reaction rates among minerals and aqueous solutions. II. Rate constants, effective surface area, and the hydrolysis of feldspar. *Geochimica et Cosmochimica Acta*, 48(12):2405–2432.

Hindmarsh, A. C., Brown, P. N., Grant, K. E., Lee, S. L., Serban, R., Shumaker, D. E., and Woodward, C. S. (2005). SUNDIALS: Suite of Nonlinear and Differential/Algebraic Equation Solvers. *ACM Transactions on Mathematical Software*, 31(3):363–396.

Hou, S.-X., Maitland, G. C., and Trusler, J. M. (2013a). Measurement and modeling of the phase behavior of the (carbon dioxide+water) mixture at temperatures from 298.15K to 448.15K. *The Journal of Supercritical Fluids*, 73:87–96.

Hou, S.-X., Maitland, G. C., and Trusler, J. M. (2013b). Phase equilibria of (CO2+H2O+NaCl) and (CO2+H2O+KCl): Measurements and modeling. *The Journal of Supercritical Fluids*, 78:78–88.

Houghton, J. T., Ding, Y., Griggs, D. J., Noguer, M., van der LINDEN, P. J., Dai, X., Maskell, K., and Johnson, C. (2001). *Climate Change 2001: The Scientific Basis*, volume 881. Cambridge University Press, Cambridge, UK.

Huff, V. N., Gordon, S., and Morrell, V. E. (1951). General method and thermodynamic tables for computation of equilibrium composition and temperature of chemical reactions. Technical Report NACA-TR-1037, National Advisory Committee for Aeronautics, USA.

IPCC (2005). *IPCC Special Report on Carbon Dioxide Capture and Storage. Prepared by Working Group III of the Intergovernmental Panel on Climate Change*. Cambridge University Press, Cambridge, United Kingdom/New York, NY, USA.

IPCC (2013). *Climate Change 2013: The Physical Science Basis. Contribution of Working Group I to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change*. Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA.

Johnson, J. W., Oelkers, E. H., and Helgeson, H. C. (1992). SUPCRT92: A software package for calculating the standard molal thermodynamic properties of minerals, gases, aqueous

species, and reactions from 1 to 5000 bar and 0 to 1000 C. *Computers & Geosciences*, 18(7):899–947.

Karpov, I., Chudnenko, K., and Kulik, D. (1997). Modeling chemical mass transfer in geochemical processes; thermodynamic relations, conditions of equilibria and numerical algorithms. *American Journal of Science*, 297(8):767–806.

Karpov, I., Chudnenko, K., Kulik, D., Avchenko, O., and Bychinski, V. (2001). Minimization of Gibbs free energy in geochemical systems by convex programming. *Geochemistry International*, 39(11):1108–1119.

Karpov, I. K., Chudnenko, K. V., Kulik, D. A., and Bychinskii, V. A. (2002). The convex programming minimization of five thermodynamic potentials other than Gibbs energy in geochemical modeling. *American Journal of Science*, 302(4):281–311.

Kelley, C. T. (1995). *Iterative Methods for Linear and Nonlinear Equations*. SIAM.

Kharaka, Y., Cole, D., Hovorka, S., Gunter, W., Knauss, K., and Freifeld, B. (2006). Gas-water-rock interactions in Frio Formation following CO2 injection: Implications for the storage of greenhouse gases in sedimentary basins. *Geology*, 34(7):577.

Kharaka, Y. K., Gunter, W. D., Aggarwal, P. K., and Perkins E., D. J. D. (1988). SOLMINEQ.88: a computer program for geochemical modeling of water-rock interactions. *USGS Water-Resources Investigations Report*, 88(4227).

Kulik, D. (2006). Dual-thermodynamic estimation of stoichiometry and stability of solid solution end members in aqueous–solid solution systems. *Chemical Geology*, 225(3-4):189–212.

Kulik, D., Berner, U., and Curti, E. (2004). Modelling chemical equilibrium partitioning with the GEMS-PSI code. Technical report, Paul Scherrer Institut, Villigen, Switzerland.

Kulik, D. A., Wagner, T., Dmytrieva, S. V., Kosakowski, G., Hingerl, F. F., Chudnenko, K. V., and Berner, U. R. (2013). GEM-Selektor geochemical modeling package: revised algorithm and GEMS3K numerical kernel for coupled simulation codes. *Computational Geosciences*, 17(1):1–24.

Kumar, A., Noh, M., Pope, G., Sepehrnoori, K., Bryant, S., and Lake, L. (2004). Reservoir Simulation of CO2 Storage in Deep Saline Aquifers. In *Proceedings of SPE/DOE Fourteenth Symposium on Improved Oil Recovery*. Society of Petroleum Engineers.

Lagneau, V., Pipart, A., and Catalette, H. (2005). Reactive Transport Modelling of CO2 Sequestration in Deep Saline Aquifers. *Science And Technology*, 60(2):231–247.

Langmuir, D. (1996). *Aqueous Environmental Geochemistry*. Prentice Hall, 1 edition.

Lasaga, A. (1998). *Kinetic Theory in the Earth Sciences*. Princeton University Press.

Lasaga, A. C. (1981). Rate laws of chemical reactions. In *Kinetics of Geochemical Processes*, chapter 1, pages 1–66. Reviews in Mineralogy and Geochemistry.

Leal, A. M., Blunt, M. J., and LaForce, T. C. (2013). A robust and efficient numerical method for multiphase equilibrium calculations: Application to CO2–brine–rock systems at high temperatures, pressures and salinities. *Advances in Water Resources*, 62(C):409–430.

Leamnson, R., Thomas, J., and Ehrlinger, H. (1969). A study of the surface areas of particulate microcrystalline silica and silica sand. *Illinois State Geological Survey*, 444.

Lichtner, P. (1985). Continuum model for simultaneous chemical reactions and mass transport in hydrothermal systems. *Geochimica et Cosmochimica Acta*, 49(3):779–800.

Lindeberg, E. and Wessel-Berg, D. (1997). Vertical convection in an aquifer column under a gas cap of CO2. *Energy Conversion and Management*, 38:S229–S234.

Logg, A., Mardal, K.-A., and Wells, G. (2011). *Automated Solution of Differential Equations by the Finite Element Method: The Fenics Book*. Lecture Notes in Computational Science and Engineering. Springer-Verlag Berlin.

Logg, A. and Wells, G. N. (2010). DOLFIN: Automated finite element computing. *ACM Transactions on Mathematical Software*, 37(2):1–28.

Lu, C. and Lichtner, P. C. (2005). PFLOTRAN: Massively Parallel 3D Simulator for CO2 Sequestration in Geologic Media. In *DOE-NETL Fourth Annual Conference on Carbon Capture and Sequestration*.

Mironenko, M. V. and Zolotov, M. Y. (2011). Equilibrium-kinetic model of water-rock interaction. *Geochemistry International*, 50(1):1–7.

Morel, F. and Morgan, J. (1972). Numerical method for computing equilibriums in aqueous chemical systems. *Environmental Science & Technology*, 6(1):58–67.

Murty, K. and Kabadi, S. (1987). Some NP-complete problems in quadratic and nonlinear programming. *Mathematical programming*, 39(2):117–129.

Nocedal, J., Wächter, A., and Waltz, R. A. (2009). Adaptive Barrier Update Strategies for Nonlinear Interior Methods. *SIAM Journal on Optimization*, 19(4):1674–1693.

Nocedal, J. and Wright, S. J. (1999). *Numerical Optimization*. Springer, 2 edition.

Nordstrom, D., Plummer, L. N., Wigley, T., Wolery, T., Ball, J., Jenne, E., Bassett, R., Crerar, D., Florence, T., Fritz, B., Hoffman, M., Holdren, G., Jr., L. G., Mattigod, S., McDuff, R., Morel, F., Reddy, M., and Sposito, G. (1979). A comparison of computerized chemical models for equilibrium calculations in aqueous systems. *American Chemical Society Symposium Series*, 93:857–892.

Obi, E.-O. I. and Blunt, M. J. (2006). Streamline-based simulation of carbon dioxide storage in a North Sea aquifer. *Water Resources Research*, 42(3):1–13.

Ormerod, B. (1994). The disposal of carbon dioxide from fossil fuel fired power stations. *IEA Greenhouse Gas R&D Programme*, Technical Rep. IEAGHG/SR3.

Palandri, J. L. and Kharaka, Y. K. (2004). A compilation of rate parameters of water-mineral interaction kinetics for application to geochemical modeling. Technical report, National Energy Technology Laboratory - United States Department of Energy, Menlo Park, California.

Parkhurst, D. and Appelo, C. (1999). User's guide to PHREEQC (Version 2)—A computer program for speciation, batch-reaction, one-dimensional transport, and inverse geochemical calculations. *USGS Water-Resources Investigations Report*, 99(4259):326.

Parkhurst, D. and Appelo, C. (2013). Description of input and examples for PHREEQC version 3—A computer program for speciation, batch-reaction, one-dimensional transport, and inverse geochemical calculations. In *Groundwater Book 6, Modeling Techniques*, chapter A43, page 497. U.S. Geological Survey Techniques and Methods.

Peng, C., Crawshaw, J. P., Maitland, G. C., Martin Trusler, J., and Vega-Maza, D. (2013). The pH of CO2-saturated water at temperatures between 308K and 423K at pressures up to 15MPa. *The Journal of Supercritical Fluids*, 82:129–137.

Peng, C., Crawshaw, J. P., Maitland, G. C., and Trusler, J. P. M. (2014). Kinetics of Calcite Dissolution in CO2-Saturated Water at Temperatures between (323 and 373) K and Pressures up to 13.8 MPa. pages 1–25. Manuscript submitted for publication.

Peng, D.-Y. and Robinson, D. B. (1976). A New Two-Constant Equation of State. *Industrial & Engineering Chemistry Fundamentals*, 15(1):59–64.

Pentland, C. H., El-Maghraby, R., Iglauer, S., and Blunt, M. J. (2011). Measurements of the capillary trapping of super-critical carbon dioxide in Berea sandstone. *Geophysical Research Letters*, 38(6):n/a–n/a.

Perez, R., Heidemann, R., and Perez, E. (2012). A new approach to multiphase geochemical speciation modeling. *Applied Geochemistry*, 27(9):1724–1737.

Perkins, E., Gunter, W., Nesbtt, H., and St-Arnaud, L. (1997). Critical review of classes of geochemical computer models adaptable for prediction of acidic drainage from mine waste rock. In *Fourth International Conference on Acid Rock Drainage*, pages 587–601, Ottawa, Ontario, Canada.

Poling, B. E., Prausnitz, J. M., and O'Connell, J. P. (2001). *The properties of gases and liquids*. McGraw-Hill.

Pruess, K., Xu, T., Apps, J., and Garcia, J. (2003). Numerical Modeling of Aquifer Disposal of CO2. *SPE Journal*, 8(1):49.

Reed, M. H. (1982). Calculation of multicomponent chemical equilibria and reaction processes in systems involving minerals, gases and an aqueous phase. *Geochimica et Cosmochimica Acta*, 46(4):513–528.

Rumpf, B., Nicolaisen, H., Ocal, C., and Maurer, G. (1994). Solubility of carbon dioxide in aqueous solutions of sodium chloride: Experimental results and correlation. *Journal of Solution Chemistry*, 23(3):431–448.

Schecher, W. and McAvoy, D. (1998). MINEQL+: A chemical equilibrium modeling system. *Environmental Research Software: Hallowell*.

Schultz, L., Andersson, M., Dalby, K., Müter, D., Okhrimenko, D., Fordsmand, H., and Stipp, S. (2013). High surface area calcite. *Journal of Crystal Growth*, 371:34–38.

Shao, H., Dmytrieva, S. V., Kolditz, O., Kulik, D. a., Pfingsten, W., and Kosakowski, G. (2009). Modeling reactive transport in non-ideal aqueous–solid solution system. *Applied Geochemistry*, 24(7):1287–1300.

Shock, E. and Helgeson, H. C. (1988). Calculation of the thermodynamic and transport properties of aqueous species at high pressures and temperatures: Correlation algorithms for ionic species and equation of state predictions to 5 kb and 1000°C. *Geochimica et Cosmochimica Acta*, 52(8):2009–2036.

Shock, E. L., Oelkers, E. H., Johnson, J. W., Sverjensky, D. A., and Helgeson, H. C. (1992). Calculation of the thermodynamic properties of aqueous species at high pressures and temperatures. Effective electrostatic radii, dissociation constants and standard partial molal properties to 1000 °C and 5 kbar. *Journal of the Chemical Society, Faraday Transactions*, 88(6):803.

Shvarov, Y. V. (1999). Algorithmization of the numeric equilibrium modeling of dynamic geochemical processes. *Geochemistry International*, 37:562–570.

Shvarov, Y. V. (2008). HCh: New potentialities for the thermodynamic simulation of geochemical systems offered by windows. *Geochemistry International*, 46(8):834–839.

Silva, R., Ulbrich, M., Ulbrich, S., and Vicente, L. N. (2008). A globally convergent primal-dual interior-point filter method for nonlinear programming: new filter optimality measures and computational results. Technical report, Department of Mathematics, University of Coimbra.

Smith, W. R. and Missen, R. W. (1982). *Chemical Reaction Equilibrium Analysis: Theory and Algorithms*. Wiley-Interscience, New York.

Spycher, N. and Pruess, K. (2005). CO2–H2O mixtures in the geological sequestration of CO2. II. Partitioning in chloride brines at 12–100 °C and up to 600 bar. *Geochimica et Cosmochimica Acta*, 69(13):3309–3320.

Spycher, N., Pruess, K., and Ennis-King, J. (2003). CO2–H2O mixtures in the geological sequestration of CO2. I. Assessment and calculation of mutual solubilities from 12 to 100°C and up to 600 bar. *Geochimica et Cosmochimica Acta*, 67(16):3015–3031.

Spycher, N. and Reed, M. (1988). Fugacity coefficients of H2, CO2, CH4, H2O and of H2O–CO2–CH4 mixtures: A virial equation treatment for moderate pressures and temperatures applicable to calculations of hydrothermal boiling. *Geochimica et Cosmochimica Acta*, 52(3):739–749.

Staudt, A., Huddleston, N., and Kraucunas, I. (2008). *Understanding and Responding to Climate Change: Highlights of National Academies Reports*. Number 0808. National Academies Press.

Steefel, C., Depaolo, D., and Lichtner, P. (2005). Reactive transport modeling: An essential tool and a new research approach for the Earth sciences. *Earth and Planetary Science Letters*, 240(3-4):539–558.

Steefel, C. and Lasaga, A. (1994). A coupled model for transport of multiple chemical species and kinetic precipitation/dissolution reactions with applications to reactive flow in single phase hydrothermal systems. *American Journal of Science*, 294(5):529–592.

Steefel, C. I. (2009). *CrunchFlow: Software for Modeling Multicomponent Reactive Flow and Transport – User's Manual*. Earth Sciences Division, Lawrence Berkeley National Laboratory, Berkeley, USA.

Steefel, C. I. and Cappellen, P. V. (1990). A new kinetic approach to modeling water-rock interaction: The role of nucleation, precursors, and Ostwald ripening. *Geochimica et Cosmochimica Acta*, 54(10):2657–2677.

Steefel, C. I. and MacQuarrie, K. T. B. (1996). Approaches to modeling of reactive transport in porous media. *Reviews in Mineralogy and Geochemistry*, 34(1):83–129.

Tanger, J. C. and Helgeson, H. C. (1988). Calculation of the thermodynamic and transport properties of aqueous species at high pressures and temperatures; revised equations of state for the standard partial molal properties of ions and electrolytes. *American Journal of Science*, 288(1):19–98.

Tong, D., Trusler, J. P. M., and Vega-Maza, D. (2013). Solubility of CO2 in Aqueous Solutions of CaCl2 or MgCl2 and in a Synthetic Formation Brine at Temperatures up to 423 K and Pressures up to 40 MPa. *Journal of Chemical & Engineering Data*, 58(7):2116–2124.

Truesdell, A. H. and Jones, B. F. (1974). WATEQ–A computer program for calculating chemical equilibrium of natural waters. *U.S. Geological Survey, Journal of Research*, 2(2):233–248.

Ulbrich, M., Ulbrich, S., and Vicente, L. N. (2004). A globally convergent primal-dual interior-point filter method for nonlinear programming. *Mathematical Programming*, 100(2):379–410.

van der Lee, J. and Windt, L. D. (2002). CHESS Tutorial and Cookbook, Version 3.0. Technical Report 1-2, Ecole des Mines de Paris, Fontainebleau, France.

Van Zeggeren, F. and Storey, S. H. (1970). *The Computation of Chemical Equilibria*. Cambridge University Press, London, England.

Vanderbei, R. and Shanno, D. (1999). An interior-point algorithm for nonconvex nonlinear programming. *Computational Optimization and Applications*, 252:231–252.

Vanderbei, R. J. (1999). LOQO: An interior point code for quadratic programming. *Optimization Methods and Software*, 11(1-4):451–484.

Vanderbei, R. J. (2006). LOQO User's Manual – Version 4.05.

Wächter, A. and Biegler, L. T. (2005a). Line Search Filter Methods for Nonlinear Programming: Local Convergence. *SIAM Journal on Optimization*, 16(1):32–48.

Wächter, A. and Biegler, L. T. (2005b). Line Search Filter Methods for Nonlinear Programming: Motivation and Global Convergence. *SIAM Journal on Optimization*, 16(1):1–31.

Wächter, A. and Biegler, L. T. (2005c). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57.

Wagner, T., Kulik, D. a., Hingerl, F. F., and Dmytrieva, S. V. (2012). GEM-Selektor Geochemical Modeling Package: TSolMod Library and Data Interface for Multicomponent Phase Models. *The Canadian Mineralogist*, 50(5):1173–1195.

Wagner, W. and Pruss, A. (2002). The IAPWS Formulation 1995 for the Thermodynamic Properties of Ordinary Water Substance for General and Scientific Use. *Journal of Physical and Chemical Reference Data*, 31(2):387.

Westall, J., Zachary, J., and Morel, F. (1976). MINEQL: A computer program for the calculation of chemical equilibrium composition of aqueous systems.

Whitson, C. and Michelsen, M. (1989). The Negative Flash. *Fluid Phase Equilibria*, 53(2):51–71.

Wolery, T. J. (1992a). EQ3/6, A Software Package for Geochemical Modeling of Aqueous Systems: Package Overview and Installation Guide. Technical report, Lawrence Livermore National Laboratory, Livermore, California.

Wolery, T. J. (1992b). EQ3NR, A Computer Program for Geochemical Squeous Speciation-Solubility Calculations: Theoretical Manual, User's Guide, and Related Documentation (Version 7.0). Technical Report UCRL-MA-110662 PT III, Lawrence Livermore National Laboratory, Livermore, California.

Wolery, T. J. and Daveler, S. A. (1992). EQ6, A Computer Program for Reaction Path Modeling of Aqueous Geochemical Systems: Theoretical Manual, User's Guide, and Related Documentation (Version 7.0).

Xu, T., Apps, J., Pruess, K., and Yamamoto, H. (2007). Numerical modeling of injection and mineral trapping of CO2 with H2S and SO2 in a sandstone formation. *Chemical Geology*, 242(3-4):319–346.

Xu, T., Apps, J. A., and Pruess, K. (2003). Reactive geochemical transport simulation to study mineral trapping for CO2 disposal in deep arenaceous formations. *Journal of Geophysical Research*, 108(B2):13.

Xu, T., Sonnenthal, E., Spycher, N., and Pruess, K. (2004). TOUGHREACT - User's Guide: A Simulation Program for Non-isothermal Multiphase Reactive geochemical Transport in Variable Saturated Geologic Media. Technical Report 2, Lawrence Berkeley National Laboratory.

Xu, T., Sonnenthal, E., Spycher, N., and Pruess, K. (2006). TOUGHREACT - A simulation program for non-isothermal multiphase reactive geochemical transport in variably saturated geologic media: Applications to geothermal injectivity and CO2 geological sequestration. *Computers & Geosciences*, 32(2):145–165.

Zeleznik, F. J. and Gordon, S. (1960). An analytical investigation of the general methods of calculating chemical equilibrium compositions. (NASA-TN-D-473).

Zeleznik, F. J. and Gordon, S. (1968). Calculation of complex chemical equilibria. *Industrial and Engineering Chemistry*, 60:27–57.

Zhang, Y. (2008). *Geochemical Kinetics*. Princeton University Press.

Zhu, C. (2009). Geochemical Modeling of Reaction Paths and Geochemical Reaction Networks. *Reviews in Mineralogy and Geochemistry*, 70(1):533–569.

Zhu, C. and Anderson, G. (2002). *Environmental Applications of Geochemical Modeling*. Cambridge University Press.

APPENDIX A

CONVERGENCE RATES OF A STOICHIOMETRIC ALGORITHM

A common stoichiometric formulation for equilibrium calculations is described here, followed by comments on its convergence rate. Different from a non-stoichiometric formulation, which is based in the minimisation of the Gibbs free energy, the stoichiometric formulation is based on the solution of a system of mass-action and mass-balance equations (Smith and Missen, 1982).

A.1 Mathematical Formulation

In what follows, the chemical system has been partitioned into primary and secondary species. The i-th primary species is denoted by α_i, and the j-th secondary species by α_j. As it is commonly done in the geochemistry literature, the subscripts i and j are enough to tell if a species is primary or secondary (Bethke, 2007). In addition, the number of primary and secondary species are denoted respectively by N_i and N_j. For simplicity reasons, only an aqueous phase is assumed in the system.

In the stoichiometric approach, the primary and secondary species are related to each other according to the following linearly independent system of equilibrium reactions:

$$\alpha_j \rightleftharpoons \sum_{i=1}^{N_i} \nu_{ji}\alpha_i \qquad (j = 1, \ldots, N_j), \tag{A.1}$$

where ν_{ji} is the stoichiometric coefficient of the i-th primary species in the j-th secondary species. For example, for the system H_2O–CO_2, we can write the following linearly independent reactions:

$$\begin{aligned} OH^- &\rightleftharpoons H_2O - H^+, \\ CO_2(aq) &\rightleftharpoons HCO_3^- + H^+ - H_2O, \end{aligned}$$

where the sets of primary and secondary species are $\{H_2O, H^+, HCO_3^-\}$ and $\{OH^-, CO_2(aq)\}$.

Computing the equilibrium state of the system requires solving the following mass-balance equations:

$$m_i + \sum_{j=1}^{N_j} \nu_{ji} m_j = M_i \qquad (i = 1, \ldots, N_i), \tag{A.2}$$

and the following mass-action equations:

$$m_j = \frac{1}{K_j \gamma_j} \prod_{i=1}^{N_i} (m_i \gamma_i)^{\nu_{ji}} \qquad (j = 1, \ldots, N_j), \tag{A.3}$$

where m_i and m_j are the molalities of the i-th primary species and j-th secondary species; M_i is the given total molality of the i-th primary species; γ_i and γ_j are the activity coefficients of the i-th primary species and j-th secondary species; and K_j is the equilibrium constant of the j-th reaction.

Remark. Using molalities as unknowns has the inconvenience of the need to handle different equations for solvent water and solutes, since the molality of water does not make sense. This discussion, however, is kept simple by listing only those mass-balance and mass-action equations that involves only solutes. The presentation, however, is not compromised by this lack of completeness.

A.2 Numerical Method

The common practice of solving equations (A.2) and (A.3) is to eliminate the dependency of the mass-balance equations (A.2) on the molalities of the secondary species m_j. This results in the following system of equations:

$$m_i + \sum_{j=1}^{N_j} \frac{\nu_{ji}}{K_j \gamma_j} \prod_{i=1}^{N_i} (m_i \gamma_i)^{\nu_{ji}} = M_i \qquad (i = 1, \ldots, N_i). \tag{A.4}$$

Let k denote the iteration counter of the calculation. Assume that the molalities of the primary and secondary species m_i^k and m_j^k are known at iteration k. Therefore, the new molalities of the primary species m_i^{k+1} can be calculated by applying Newton's method to the following system of equations:

$$m_i^{k+1} + \sum_{j=1}^{N_j} \frac{\nu_{ji}}{K_j \gamma_j^k} \prod_{i=1}^{N_i} (m_i^{k+1} \gamma_i^k)^{\nu_{ji}} = M_i \qquad (i = 1, \ldots, N_i). \tag{A.5}$$

Then, using the new values m_i^{k+1}, we can update the molalities of the secondary species as:

$$m_j^{k+1} = \frac{1}{K_j \gamma_j^k} \prod_{i=1}^{N_i} (m_i^{k+1} \gamma_i^k)^{\nu_{ji}} \qquad (j = 1, \ldots, N_j). \tag{A.6}$$

where it can be observed the resemblance of this update of m_j^{k+1} with a successive substitution method.

Note that the dependence of the activity coefficients γ_i and γ_j on the molalities m_i and m_j is commonly ignored, and their values are hold constant until the end of the iteration. According to Bethke (2007), this is called the *soft formulation*. The *hard formulation*, on the other hand, takes into account a partial dependence of the activity coefficients γ_i and γ_j on the molalities of the

primary species m_i only. Their update is done once both m_i^{k+1} and m_j^{k+1} have been obtained, after which the previous two-stage calculations using equations (A.5) and (A.6) are repeated until convergence.

If a full dependence of the activity coefficients on the species molalities were taken into account, then equation (A.5) would become:

$$m_i^{k+1} + \sum_{j=1}^{N_j} \frac{\nu_{ji}}{K_j \gamma_j^{k+1}} \prod_{i=1}^{N_i} (m_i^{k+1} \gamma_i^{k+1})^{\nu_{ji}} = M_i \qquad (i = 1, \ldots, N_i), \tag{A.7}$$

where γ_i^{k+1} and γ_j^{k+1} are unknowns that depend on m_i^{k+1} and m_j^{k+1}. As a result, it is no longer possible to calculate m_i^{k+1} and m_j^{k+1} in two stages as before. This formulation based on the law of mass-action approach that adopts a complete Newton scheme has been further studied in Leal et al. (2013).

A.3 Discussion

The motivation of only applying Newton's method to equation (A.5) is to avoid large linear systems. This incomplete Newton's scheme results in Jacobian matrices that are only $N_i \times N_i$, and since the number of primary species N_i is in general considerably smaller than that of secondary species N_j, the size of the linear systems is substantially decreased.

This incomplete scheme is only possible if the non-linear dependence of the activity coefficients on the molalities of the species is *completely* or *partially* ignored. Equations (A.5) and (A.6) were obtained by completely ignoring this dependence, using a soft formulation, and so the following derivative terms are not included in the Jacobian matrix:

$$\frac{\partial \gamma_i}{\partial m_i}, \ \frac{\partial \gamma_i}{\partial m_j}, \ \frac{\partial \gamma_j}{\partial m_i}, \ \frac{\partial \gamma_j}{\partial m_j}, \tag{A.8}$$

which are extremely valuable information. Its absence can negatively impact the convergence rate of the calculation, specially in strong non-ideal systems or near phase boundaries, where these values are very sensitive. If, on the other hand, a hard formulation were used to obtain equation (A.5), then only the derivative terms:

$$\frac{\partial \gamma_i}{\partial m_i}, \ \frac{\partial \gamma_j}{\partial m_i}, \tag{A.9}$$

would have been taken into account. Since in general $N_i \ll N_j$, the above terms represent just a small portion of all that should be considered.

Thus, we see that the resulting Jacobian matrix, in both the soft and hard formulations, always lacks important first-order partial derivative information. Since quadratic convergence rates in Newton's method is subject to exact Jacobian expressions, the presented stoichiometric method should expect slower convergence rates. In fact, it can be said that the incomplete Newton's method is fundamentally more similar to the family of quasi-Newton methods, which converges Q-superlinearly, than standard Newton methods, which converges Q-quadratically (see Kelley, 1995; Nocedal and Wright, 1999).

Depending on the kind of chemical system, and its physical and chemical conditions, a soft or hard stoichiometric method might still converge in few iterations. Perhaps even less than an

approach that attains quadratic convergence rates, since these optimal rates only happen near the solution. Therefore, it is near the solution that the algorithms should differ in performance, since one that converges with quadratic rates should do so in very few iterations. From this we see that for equilibrium problems in the context of chemical kinetics or reactive transport modelling, where the calculations always start with hot initial guesses, a well implemented quadratic convergent method can deliver optimal performance.

APPENDIX B

THERMODYNAMIC MODELS

In this section we present some thermodynamic models for the calculation of the activity and fugacity coefficient models for aqueous, gaseous and mineral species. We show how these coefficients can be used to compute the activities of the species, which is a thermodynamic quantity required to calculate the chemical equilibrium state of a multiphase system.

B.1 Activity Coefficients for Aqueous Species

In this section we present some thermodynamic models for activity coefficients for aqueous species. The activity coefficient γ_i of an aqueous species is assumed in this work to be on a molality scale, so that its activity a_i can be calculated as:

$$a_i = \gamma_i m_i, \tag{B.1}$$

where m_i is the molality of the i-th species given by:

$$m_i = 55.508 \frac{n_i}{n_w}, \tag{B.2}$$

with n_w denoting the number of moles of species $H_2O(l)$.

In what follows, we let I and $\bar{I}$ denote, respectively, the effective and stoichiometric ionic strength of an aqueous solution. These are given by:

$$I = \frac{1}{2} \sum_{j=1}^{N_j} m_j Z_j^2 \tag{B.3}$$

and

$$\bar{I} = \frac{1}{2} \sum_{j=1}^{N_k} m_j^* Z_j^2, \tag{B.4}$$

where N_j and N_k denote the number of ions and aqueous complexes respectively; Z_j the electrical charge of the j-th ion; and m_i^* the stoichiometric molality of the j-th ion, which is given by:

$$m_j^* = \sum_{k=1}^{N_k} \nu_{kj} m_k, \tag{B.5}$$

with ν_{kj} denoting the stoichiometry of the j-th ion in the k-th aqueous complex; and m_k the molality of the k-th aqueous complex. The stoichiometric molality of an ion is the molality this ion would have if all aqueous complexes were completely dissolved in the aqueous solution.

HKF Model for Ionic Species

The activity coefficients of the ionic species are calculated using (see Helgeson et al. 1981, equation 298):

$$\log \gamma_j = -\frac{A_\gamma Z_j^2 \sqrt{\bar{I}}}{\Lambda} + \log x_w + [\omega_j^{\text{abs}} b_{\text{NaCl}} + b_{\text{Na}^+\text{Cl}^-} - 0.19(|Z_j| - 1)]\bar{I}, \tag{B.6}$$

where x_w denotes the molar fraction of water; and Λ is the Debye-Hückel function defined by:

$$\Lambda = 1 + \text{å} B_\gamma \sqrt{\bar{I}}, \tag{B.7}$$

with å denoting the ion-size parameter given by equations (124) and (125) of Helgeson et al. (1981):

$$\text{å} = \frac{\sum_k^{N_k} \sum_j^{N_j} \nu_{kj} r_{e,j}}{\sum_k^{N_k} \sum_j^{N_j} \nu_{kj}}.$$

Both the absolute Born coefficient ω_j^{abs} and the effective electrostatic radius $r_{e,j}$ of the j-th ion are given in Table 3 of Helgeson et al. (1981).

The parameters A_γ, B_γ, b_{NaCl} and $b_{\text{Na}^+\text{Cl}^-}$ have been calculated using equations and parameters from Helgeson and Kirkham (1974a,b, 1976); Helgeson et al. (1981) and Helgeson et al. (1981). Tables B.1–B.4 present the results of this calculation over a wide range of temperatures and pressures. These tables are used in our code to interpolate those parameters at a given temperature and pressure.

The extended Debye-Hückel equation (B.6) is valid at temperatures and pressures up to 600 °C and 5000 bar. The use of the stoichiometric ionic strength $\bar{I}$ instead of the effective ionic strength I has been determined to yield more accurate activity coefficients at NaCl brine salinities up to ~6 molal.

HKF Model for Water

Helgeson et al. (1981) derived an equation for the activity of water a_w, which we present in the following form:

$$\ln a_w = \frac{2.303}{55.508} \sum_{j=1}^{N_j} m_j^* \psi_j, \tag{B.8}$$

where

$$\psi_j = \frac{A_\gamma Z_j^2 \sqrt{\bar{I}}}{3}\sigma + \frac{x_w}{1 - x_w} \log x_w - \frac{1}{2}[\omega_j b_{\text{NaCl}} + b_{\text{Na}^+\text{Cl}^-} - 0.19(|Z_j| - 1)]\bar{I} \tag{B.9}$$

and

$$\sigma = \frac{3}{(\text{å} B_\gamma \sqrt{\bar{I}})^3} \left(\Lambda - \frac{1}{\Lambda} - 2 \ln \Lambda \right). \tag{B.10}$$

Table B.1: The activity coefficient parameter A_γ of the HKF model in units of $kg^{1/2}/mol^{1/2}$ calculated at temperatures 0–500 °C and pressures up to 5000 bar.

T (°C)	P (bar)									
	P_{sat}	250	500	750	1000	1500	2000	3000	4000	5000
0	0.4939	0.4871	0.4810	0.4755	0.4705	0.4617	0.4544	0.4427	0.4337	0.4266
25	0.5114	0.5047	0.4985	0.4928	0.4875	0.4782	0.4701	0.4568	0.4462	0.4375
50	0.5354	0.5281	0.5213	0.5151	0.5094	0.4991	0.4901	0.4750	0.4628	0.4526
75	0.5649	0.5565	0.5488	0.5418	0.5353	0.5236	0.5134	0.4963	0.4824	0.4707
100	0.5996	0.5897	0.5807	0.5725	0.5649	0.5515	0.5398	0.5202	0.5043	0.4910
125	0.6396	0.6276	0.6168	0.6070	0.5981	0.5823	0.5688	0.5463	0.5282	0.5131
150	0.6855	0.6707	0.6573	0.6455	0.6348	0.6161	0.6003	0.5743	0.5536	0.5365
175	0.7383	0.7195	0.7028	0.6881	0.6751	0.6528	0.6341	0.6040	0.5803	0.5609
200	0.7995	0.7753	0.7538	0.7355	0.7194	0.6925	0.6703	0.6352	0.6081	0.5861
225	0.8718	0.8401	0.8117	0.7882	0.7682	0.7353	0.7088	0.6678	0.6367	0.6118
250	0.9596	0.9169	0.8783	0.8476	0.8221	0.7815	0.7498	0.7018	0.6662	0.6381
275	1.0704	1.0111	0.9563	0.9152	0.8823	0.8317	0.7934	0.7372	0.6964	0.6646
300	1.2183	1.1325	1.0500	0.9934	0.9502	0.8865	0.8400	0.7739	0.7272	0.6915
325	1.4357	1.3019	1.1668	1.0856	1.0277	0.9467	0.8900	0.8122	0.7588	0.7186
350	1.8233	1.5767	1.3188	1.1970	1.1175	1.0132	0.9438	0.8521	0.7910	0.7460
375		2.2948	1.5300	1.3350	1.2230	1.0871	1.0019	0.8937	0.8241	0.7737
400			1.8509	1.5114	1.3490	1.1700	1.0648	0.9373	0.8581	0.8019
425			2.3997	1.7439	1.5014	1.2632	1.1333	0.9831	0.8930	0.8305
450			3.3844	2.0579	1.6877	1.3685	1.2078	1.0311	0.9291	0.8597
475			4.7805	2.4810	1.9161	1.4874	1.2889	1.0815	0.9662	0.8895
500			6.0949	3.0235	2.1930	1.6213	1.3770	1.1344	1.0047	0.9202

Table B.2: The activity coefficient parameter B_γ of the HKF model in units of $kg^{1/2}/(mol^{1/2} \cdot cm)$ calculated at temperatures 0–500 °C and pressures up to 5000 bar.

T (°C)	P (bar)									
	P_{sat}	250	500	750	1000	1500	2000	3000	4000	5000
0	0.4939	0.4871	0.4810	0.4755	0.4705	0.4617	0.4544	0.4427	0.4337	0.4266
25	0.5114	0.5047	0.4985	0.4928	0.4875	0.4782	0.4701	0.4568	0.4462	0.4375
50	0.5354	0.5281	0.5213	0.5151	0.5094	0.4991	0.4901	0.4750	0.4628	0.4526
75	0.5649	0.5565	0.5488	0.5418	0.5353	0.5236	0.5134	0.4963	0.4824	0.4707
100	0.5996	0.5897	0.5807	0.5725	0.5649	0.5515	0.5398	0.5202	0.5043	0.4910
125	0.6396	0.6276	0.6168	0.6070	0.5981	0.5823	0.5688	0.5463	0.5282	0.5131
150	0.6855	0.6707	0.6573	0.6455	0.6348	0.6161	0.6003	0.5743	0.5536	0.5365
175	0.7383	0.7195	0.7028	0.6881	0.6751	0.6528	0.6341	0.6040	0.5803	0.5609
200	0.7995	0.7753	0.7538	0.7355	0.7194	0.6925	0.6703	0.6352	0.6081	0.5861
225	0.8718	0.8401	0.8117	0.7882	0.7682	0.7353	0.7088	0.6678	0.6367	0.6118
250	0.9596	0.9169	0.8783	0.8476	0.8221	0.7815	0.7498	0.7018	0.6662	0.6381
275	1.0704	1.0111	0.9563	0.9152	0.8823	0.8317	0.7934	0.7372	0.6964	0.6646
300	1.2183	1.1325	1.0500	0.9934	0.9502	0.8865	0.8400	0.7739	0.7272	0.6915
325	1.4357	1.3019	1.1668	1.0856	1.0277	0.9467	0.8900	0.8122	0.7588	0.7186
350	1.8233	1.5767	1.3188	1.1970	1.1175	1.0132	0.9438	0.8521	0.7910	0.7460
375		2.2948	1.5300	1.3350	1.2230	1.0871	1.0019	0.8937	0.8241	0.7737
400			1.8509	1.5114	1.3490	1.1700	1.0648	0.9373	0.8581	0.8019
425			2.3997	1.7439	1.5014	1.2632	1.1333	0.9831	0.8930	0.8305
450			3.3844	2.0579	1.6877	1.3685	1.2078	1.0311	0.9291	0.8597
475			4.7805	2.4810	1.9161	1.4874	1.2889	1.0815	0.9662	0.8895
500			6.0949	3.0235	2.1930	1.6213	1.3770	1.1344	1.0047	0.9202

Note: The values of B_γ were multiplied by 10^{-8}.

Table B.3: The activity coefficient parameter b_{NaCl} of the HKF model in units of kg/cal calculated at temperatures 0–500 °C and pressures up to 5000 bar.

T (°C)	P (bar)									
	P_{sat}	250	500	750	1000	1500	2000	3000	4000	5000
0	21.962	22.211	22.437	22.643	22.831	23.162	23.444	23.901	24.258	24.548
25	18.081	18.321	18.542	18.746	18.934	19.273	19.569	20.063	20.461	20.792
50	14.530	14.783	15.016	15.232	15.432	15.794	16.112	16.648	17.088	17.458
75	11.235	11.516	11.775	12.012	12.233	12.630	12.979	13.570	14.055	14.465
100	8.125	8.449	8.745	9.015	9.264	9.710	10.100	10.757	11.295	11.749
125	5.138	5.521	5.868	6.183	6.470	6.980	7.421	8.158	8.757	9.260
150	2.214	2.678	3.095	3.467	3.804	4.395	4.900	5.733	6.402	6.960
175	-0.710	-0.137	0.376	0.826	1.227	1.919	2.503	3.449	4.200	4.821
200	-3.703	-2.981	-2.336	-1.783	-1.298	-0.477	0.201	1.283	2.127	2.817
225	-6.858	-5.930	-5.096	-4.402	-3.807	-2.824	-2.028	-0.787	0.164	0.932
250	-10.304	-9.082	-7.969	-7.080	-6.339	-5.147	-4.209	-2.779	-1.707	-0.852
275	-14.247	-12.590	-11.042	-9.873	-8.934	-7.476	-6.362	-4.709	-3.498	-2.547
300	-19.060	-16.716	-14.437	-12.856	-11.643	-9.837	-8.506	-6.592	-5.223	-4.166
325	-25.556	-21.993	-18.343	-16.122	-14.524	-12.262	-10.663	-8.439	-6.893	-5.718
350	-36.227	-29.865	-23.076	-19.806	-17.650	-14.783	-12.853	-10.265	-8.519	-7.214
375		-48.639	-29.220	-24.100	-21.113	-17.436	-15.095	-12.080	-10.108	-8.663
400			-38.002	-29.287	-25.032	-20.261	-17.410	-13.895	-11.671	-10.072
425			-52.214	-35.791	-29.554	-23.299	-19.818	-15.719	-13.214	-11.449
450			-76.652	-44.209	-34.857	-26.594	-22.338	-17.561	-14.746	-12.801
475			-110.794	-55.166	-41.133	-30.187	-24.986	-19.429	-16.271	-14.134
500			-143.763	-68.878	-48.527	-34.113	-27.778	-21.329	-17.796	-15.455

Note: The values of b_{NaCl} were multiplied by 10^7.

Table B.4: The activity coefficient parameter $b_{Na^+Cl^-}$ of the HKF model in units of kg/mol calculated at temperatures 0–500 °C and pressures up to 5000 bar.

T (°C)	P (bar)									
	P_{sat}	250	500	750	1000	1500	2000	3000	4000	5000
0	-15.448	-14.872	-14.390	-14.002	-13.708	-13.401	-13.471	-14.739	-17.512	-21.789
25	-9.752	-9.563	-9.404	-9.276	-9.178	-9.073	-9.090	-9.487	-10.370	-11.739
50	-5.630	-5.603	-5.579	-5.560	-5.544	-5.524	-5.518	-5.552	-5.647	-5.801
75	-2.411	-2.466	-2.510	-2.546	-2.571	-2.594	-2.577	-2.430	-2.128	-1.672
100	0.244	0.145	0.063	-0.002	-0.051	-0.097	-0.075	0.172	0.691	1.482
125	2.529	2.405	2.301	2.218	2.156	2.097	2.122	2.426	3.069	4.052
150	4.559	4.421	4.305	4.212	4.142	4.074	4.101	4.438	5.153	6.246
175	6.406	6.263	6.139	6.040	5.966	5.894	5.921	6.276	7.032	8.187
200	8.119	7.976	7.848	7.746	7.670	7.595	7.623	7.987	8.763	9.950
225	9.729	9.591	9.462	9.359	9.282	9.205	9.233	9.600	10.384	11.583
250	11.259	11.130	11.001	10.897	10.820	10.743	10.771	11.138	11.921	13.119
275	12.723	12.608	12.479	12.377	12.300	12.224	12.251	12.614	13.391	14.581
300	14.133	14.036	13.909	13.807	13.731	13.656	13.682	14.041	14.808	15.984
325	15.496	15.421	15.296	15.196	15.121	15.047	15.073	15.426	16.182	17.339
350	16.818	16.771	16.648	16.550	16.476	16.403	16.428	16.775	17.517	18.654
375		18.090	17.969	17.872	17.800	17.728	17.753	18.093	18.821	19.935
400		19.380	19.262	19.167	19.096	19.026	19.050	19.383	20.096	21.188
425		20.645	20.529	20.437	20.367	20.298	20.322	20.648	21.346	22.415
450		21.887	21.774	21.683	21.615	21.547	21.570	21.890	22.572	23.618
475		23.108	22.997	22.908	22.841	22.775	22.798	23.110	23.777	24.801
500		24.308	24.199	24.113	24.048	23.983	24.005	24.310	24.963	25.964

Note: The values of $b_{Na^+Cl^-}$ were multiplied by 10^2.

The above activity model is also valid at temperatures, pressures and brine salinities up to 600 °C, 5000 bar and 6 molal. The parameters A_γ, B_γ, b_{NaCl} and $b_{\mathrm{Na^+Cl^-}}$ are interpolated from Tables B.1–B.4.

Setschenow Model for Neutral Species

The activity coefficients for neutral species are calculated using the Setschenow model equation:

$$\log \gamma_i = b_i I + \log x_w, \tag{B.11}$$

where b_i is the Setschenow coefficient of the i-th species. The value $b_i = 0.1$ is adopted in this work for those neutral species whose Setschenow coefficient is not known.

Drummond (1981) Model for CO_2(aq)

Drummond (1981) derived the following equation for the activity coefficient of CO_2(aq):

$$\ln \gamma_{\mathrm{CO_2}} = \left(a_1 + a_2 T + \frac{a_3}{T}\right) I - (a_4 + a_5 T)\frac{I}{I+1}, \tag{B.12}$$

where $a_1 = -1.0312$, $a_2 = 1.2806 \cdot 10^{-3}$, $a_3 = 255.9$, $a_4 = 0.4445$ and $a_5 = -1.606 \cdot 10^{-3}$; and T is temperature in units of Kelvin. This equation is valid within the temperature and salinity ranges 20–400 °C and 0–6.5 molal respectively.

Duan and Sun (2003) Model for CO_2(aq)

Duan and Sun (2003) presents the following activity coefficient model for CO_2(aq):

$$\begin{aligned}\ln \gamma_{\mathrm{CO_2}} = {} & 2\lambda(m_{\mathrm{Na^+}} + m_{\mathrm{K^+}} + 2m_{\mathrm{Ca^{2+}}} + 2m_{\mathrm{Mg^{2+}}}) \\ & + \zeta(m_{\mathrm{Na^+}} + m_{\mathrm{K^+}} + m_{\mathrm{Ca^{2+}}} + m_{\mathrm{Mg^{2+}}})m_{\mathrm{Cl^-}} \\ & - 0.07 m_{\mathrm{SO_4^{2-}}}\end{aligned} \tag{B.13}$$

where

$$\begin{aligned}\lambda = {} & -0.411370585 + 6.07632013 \cdot 10^{-4} T \\ & + 97.5347708/T - 0.0237622469 P/T \\ & + 0.0170656236 P/(630 - T) \\ & + 1.41335834 \cdot 10^{-5} T \ln P\end{aligned} \tag{B.14}$$

and

$$\begin{aligned}\zeta = {} & 3.36389723 \cdot 10^{-4} - 1.98298980 \cdot 10^{-5} T \\ & + 2.12220830 \cdot 10^{-3} P/T \\ & - 5.24873303 \cdot 10^{-3} P/(630 - T),\end{aligned} \tag{B.15}$$

with temperature T and pressure P given in units of Kelvin and bar respectively.

These equations are valid within the temperature, pressure and salinity ranges of 0–260 °C, 0–2000 bar, and 0–4.3 molal. Nevertheless, its use at higher salinities (e.g. up to ~6 molal NaCl and up to ~4 molal $CaCl_2$) yields satisfactory results and this has been done before in Spycher and Pruess (2005).

Rumpf et al. (1994) Model for CO_2(aq)

Based on a Pitzer formulation with correlation of solubility data, Rumpf et al. (1994) derived the following activity coefficient model for CO_2(aq) (see Spycher and Pruess 2005):

$$\begin{aligned}\ln \gamma_{CO_2} = {} & 2\mathrm{B}(m_{\mathrm{Na}^+} + m_{\mathrm{K}^+} + 2m_{\mathrm{Ca}^{2+}} + 2m_{\mathrm{Mg}^{2+}}) \\ & + 3\Gamma(m_{\mathrm{Na}^+} + m_{\mathrm{K}^+} + m_{\mathrm{Ca}^{2+}} + m_{\mathrm{Mg}^{2+}})m_{\mathrm{Cl}^-}\end{aligned} \tag{B.16}$$

where $\Gamma = -0.0028$ and

$$\mathrm{B} = 0.254 - \frac{76.82}{T} - \frac{10656}{T^2} + \frac{6312 \cdot 10^3}{T^3}, \tag{B.17}$$

with temperature T given in units of Kelvin.

These equations are valid within the temperature range of 40–160 °C, at brine salinities up to ~6 molal NaCl and at pressures up to 100 bar. Therefore, this model is unsuitable for CO_2 solubility modelling at high pressures, though it is accurate enough within its valid temperature and pressure ranges.

B.2 Fugacity Coefficients for Gaseous Species

In this section we present some thermodynamic models for fugacity coefficients for gaseous species. The fugacity coefficient φ_i of a gaseous species is used to calculate its activity a_i as:

$$a_i = \varphi_i y_i \frac{P}{P^\circ}, \tag{B.18}$$

where y_i is the molar fraction of the i-th gaseous species in the gaseous phase; and $P^\circ = 1$ bar is the standard pressure assumed in this work.

Spycher and Reed (1988) Model for H_2O(g)-CO_2(g)-CH_4(g)

Spycher and Reed (1988) derived fugacity coefficient equations for pure and mixed gases based on a virial expansion formulation. In this work we opted to use the equations of Spycher and Reed (1988) where the gaseous phase is assumed as a non-ideal mixture of gases because of its reliability and accuracy at pressures higher than the saturation pressure of water.

In what follows, we denote 1-H_2O(g), 2-CO_2(g) and 3-CH_4(g). The fugacity coefficients of the end-members of the gaseous mixture H_2O–CO_2-CH_4, therefore, are given by:

$$\begin{aligned}\ln \varphi_i = {} & \left[2\sum_{k=1}^{3} \hat{y}_k B'_{ik} - B'_{\mathrm{mix}}\right] P \\ & + \left[\frac{3}{2}\sum_{k=1}^{3}\sum_{l=1}^{3} \hat{y}_k \hat{y}_l C'_{ikl} - C'_{\mathrm{mix}}\right] P^2,\end{aligned} \tag{B.19}$$

where $\hat{y}_k$ denotes the molar fraction of the k-th gas in the mixture H_2O–CO_2-CH_4. This is in contrast with y_k, which represents the molar fraction of the k-th gas in the gaseous phase, which may contain other additional gases (e.g., O_2(g), H_2(g)). The second and third cross-virial

Table B.5: Coefficients from Table 1 of Spycher and Reed (1988) for the calculation of fugacity coefficients of pure gases.

Gas	T (°C)	P_{max} (bar)	a	b	c ($\times 10^5$)	d ($\times 10^2$)	e ($\times 10^5$)	f ($\times 10^8$)
H_2	25–600	3000	-12.5908	0.259789	-7.24730	0.471947	-2.69962	2.15622
H_2O	0–1000	1000	-3238.36	4.69231	-175.120	272.189	-463.667	202.904
	0–340	P_{sat}	-6191.41	14.8528	-914.267	-6633.26	18277.0	-13274.0
CO_2	400–1000	1000	-361.447	0.553372	-0.549789	16.3871	3.86767	-9.26594
	50-350	500	-1430.87	3.59800	-227.376	347.644	-1042.47	846.271
CH_4	16–350	500	-537.779	1.54946	-92.7827	120.861	-370.814	333.804

coefficients B'_{ij} and C'_{ijk} are calculated using:

$$B'_{ij} = \frac{a_{ij}}{T^2} + \frac{b_{ij}}{T} + c_{ij}, \tag{B.20}$$

$$C'_{ijk} = \frac{d_{ijk}}{T^2} + \frac{e_{ijk}}{T} + f_{ijk}, \tag{B.21}$$

where

$$B'_{\text{mix}} = \sum_{i=1}^{3}\sum_{k=1}^{3} \hat{y}_i \hat{y}_k B'_{ik}, \tag{B.22}$$

$$C'_{\text{mix}} = \sum_{i=1}^{3}\sum_{k=1}^{3}\sum_{l=1}^{3} \hat{y}_i \hat{y}_k \hat{y}_l C'_{ikl}. \tag{B.23}$$

The parameters a_{ij}, b_{ij}, c_{ij}, d_{ijk}, e_{ijk} and f_{ijk} are presented in Tables B.5 and B.6. Note that temperature T and pressure P are given in units of Kelvin and bar respectively.

Spycher et al. (2003) Model for H_2O(g)-CO_2(g)

From a modified Redlich-Kwong equation of state, Spycher et al. (2003) derived the following fugacity coefficient equations for the for the gaseous species H_2O(g) and CO_2(g):

$$\begin{aligned}\ln \phi_{H_2O} = &\ln\left(\frac{v}{v-b_m}\right) + \left(\frac{b_{H_2O}}{v-b_m}\right) \\ &+ \frac{a_m b_{H_2O}}{RT^{1.5} b_m}\left[\ln\left(\frac{v+b_{H_2O}}{v}\right) - \frac{b_m}{v+b_m}\right] \\ &- \frac{2a_{CO_2-H_2O}}{RT^{0.5} b_m}\ln\left(\frac{v+b_m}{v}\right) - \ln\left(\frac{Pv}{RT}\right),\end{aligned} \tag{B.24}$$

and

$$\begin{aligned}\ln \phi_{CO_2} = &\ln\left(\frac{v}{v-b_m}\right) + \left(\frac{b_{CO_2}}{v-b_m}\right) \\ &+ \frac{a_m b_{CO_2}}{RT^{1.5} b_m}\left[\ln\left(\frac{v+b_{CO_2}}{v}\right) - \frac{b_m}{v+b_m}\right] \\ &- \frac{2a_{CO_2}}{RT^{0.5} b_m}\ln\left(\frac{v+b_m}{v}\right) - \ln\left(\frac{Pv}{RT}\right)\end{aligned} \tag{B.25}$$

where v is the molar volume of the mixture, in units of cm^3/mol, given by the solution of the cubic equation:

$$v^3 - v^2\left(\frac{RT}{P}\right) - v\left(\frac{RTb_m}{P} - \frac{a_m}{PT^{0.5}} + b_m^2\right) - \frac{a_m b_m}{PT^{0.5}} = 0 \tag{B.26}$$

Table B.6: Cross-coefficients from Table 2 of Spycher and Reed (1988) for the calculation of fugacity coefficients of mixed gases.

Mixture i j	T (°C)	P_{max} (bar)	a_{ij}	b_{ij}	c_{ij} ($\times 10^2$)	d_{iij}	e_{iij} ($\times 10^2$)	f_{iij} ($\times 10^5$)	d_{ijj}	e_{ijj} ($\times 10^2$)	f_{ijj} ($\times 10^5$)
H_2O–CO_2	450–1000	1000	-1286.47	2.95028	-0.165412	2.54908	-0.494212	0.239023	0.0	0.0	0.0
	50–350	500	-1954.70	7.74805	-1.029010	104.453	-38.42830	36.58580	-8.28426	1.19097	0.808886
H_2O–CH_4	25–100	94	-1103.20	4.52871	-0.507784	0.0	0.0	0.0	0.0	0.0	0.0
CO_2–CH_4	40–240	500	-800.592	2.28990	-0.153917	2.99160	-1.04893	1.02627	1.58384	-0.492077	0.430104

Note: For H_2O–CO_2 mixtures, the coefficients of this table for the temperature range 50–350 °C must be used only with the coefficients for the range 0–340 °C in Table B.5.

Table B.7: The parameters for equations (B.24) and (B.25) taken from Spycher et al. 2003.

Parameter	Value	Unit
b_{H_2O}	18.18	cm^3/mol
b_{CO_2}	27.8	cm^3/mol
$a_{CO_2-H_2O}$	$7.89 \cdot 10^7$	$bar \cdot cm^6 \cdot K^{0.5}/mol^2$
a_{CO_2}	$7.54 \cdot 10^7 - 4.13 \cdot 10^4 T$	$bar \cdot cm^6 \cdot K^{0.5}/mol^2$

with temperature T and pressure P given in units of Kelvin and bar respectively. The parameters from the previous equations are shown in Table B.7. These equations are valid at temperatures 12–100 °C and pressures up to 600 bar.

Note that Spycher et al. (2003) assumed that $\hat{y}_{CO_2} = 1$ and $\hat{y}_{H_2O} = 0$ to derive equations (B.24) and (B.25). This simplification results in $a_m = a_{CO_2}$ and $b_m = b_{CO_2}$.

Duan et al. (2006) Model for CO_2(g)

Duan et al. (2006) presents an improved fugacity coefficient equation for CO_2(g), which can be calculated efficiently and directly as opposed to the costly and iterative scheme required in Duan and Sun (2003). The fugacity coefficient equation is given by:

$$\begin{aligned}\varphi_{CO_2} = {} & c_1 + [c_2 + c_3 T + c_4/T + c_5/(T-150)]P \\ & + (c_6 + c_7 T + c_8/T)P^2 + (c_9 + c_{10} T + c_{11}/T)\ln P \\ & + (c_{12} + c_{13} T)/P + c_{14}/T + c_{15} T^2,\end{aligned} \tag{B.27}$$

where temperature T and pressure P are given in units of Kelvin and bar respectively; and the parameters c_1 through c_{15} are given in Table B.8. This equation is valid at temperatures 0–260 °C and pressures up to 2000 bar.

In Table B.8, the auxiliary pressure P^*, in units of bar, is calculated using:

$$P^* = \begin{cases} P^{sat}_{CO_2} & T < 305 \text{ K} \\ 75 + 1.25(T - 305) & 305 \text{ K} < T < 405 \text{ K} \\ 200 & T > 405 \text{ K} \end{cases}, \tag{B.28}$$

where $P^{sat}_{CO_2}$ is the saturation pressure of CO_2, which can be computed using the equation of state developed by Poling et al. (2001):

$$\ln\left(\frac{P^{sat}_{CO_2}}{P^{cr}_{CO_2}}\right) = \frac{a_1 x + a_2 x^{1.5} + a_3 x^3 + a_4 x^6}{1 - x}, \tag{B.29}$$

with $a_1 = -6.95626$, $a_2 = 1.19695$, $a_3 = -3.12614$, $a_4 = 2.99448$ and $x = 1 - T/T^{cr}_{CO_2}$, where $T^{cr}_{CO_2} = 304.2$ K and $P^{cr}_{CO_2} = 73.83$ bar are the critical temperature and pressure of CO_2 respectively.

B.3 Activity Coefficients for Mineral Species

In this work the ideal model for the activity coefficients of mineral species is assumed. Therefore, the activity a_i of the i-th mineral in a mineral solution is given by:

$$a_i = x_i, \tag{B.30}$$

Table B.8: The parameters for the fugacity coefficient model of Duan et al. (2006) for CO_2(g).

Parameter	T–P Range					
	1	2	3	4	5	6
c_1	1	-7.1734882E-1	-6.5129019E-2	5.0383896E+0	-1.6063152E+1	-1.5693490E-1
c_2	4.7586835E-3	1.5985379E-4	-2.1429977E-4	-4.4257744E-3	-2.7057990E-3	4.4621407E-4
c_3	-3.3569963E-6	-4.9286471E-7	-1.1444930E-6	0	0	-9.1080591E-7
c_4	0	0	0	1.9572733E+0	1.4119239E-1	0
c_5	-1.3179396E+0	0	0	0	0	0
c_6	-3.8389101E-6	-2.7855285E-7	-1.1558081E-7	2.4223436E-6	8.1132965E-7	1.0647399E-7
c_7	0	1.1877015E-9	1.1952370E-9	0	0	2.4273357E-10
c_8	2.2815104E-3	0	0	-9.3796135E-4	-1.1453082E-4	0
c_9	0	0	0	-1.5026030E+0	2.3895671E+0	3.5874255E-1
c_{10}	0	0	0	3.0272240E-3	5.0527457E-4	6.3319710E-5
c_{11}	0	0	0	-3.1377342E+1	-1.7763460E+1	-2.4989661E+2
c_{12}	0	-9.6539512E+1	-2.2134306E+2	-1.2847063E+1	9.8592232E+2	0
c_{13}	0	4.4774938E-1	0	0	0	0
c_{14}	0	1.0181078E+2	7.1820393E+1	0	0	8.8876800E+2
c_{15}	0	5.3783879E-6	6.6089246E-6	-1.5056648E-5	-5.4965256E-7	-6.6348003E-7

Range 1: 273 K $< T <$ 573 K, $P < P^*$
Range 2: 273 K $< T <$ 340 K, $P^* < P <$ 1000 bar
Range 3: 273 K $< T <$ 340 K, $P >$ 1000 bar
Range 4: 340 K $< T <$ 435 K, $P^* < P <$ 1000 bar
Range 5: 340 K $< T <$ 435 K, $P >$ 1000 bar
Range 6: $T >$ 435 K, $P > P^*$

from which it follows that $a_i = 1$ for pure minerals.

APPENDIX C

DETAILS OF THE INTERIOR-POINT METHOD ALGORITHM

C.1 Central Neighbourhood: Largest Feasible Trust-Region Radius

This presents the approach for the calculation of the *largest feasible trust-region radius* Δ_k^{f} that satisfies the bound constraints:

$$\mathbf{x}_k(\Delta) \geq \mathbf{0} \qquad \text{and} \qquad \mathbf{z}_k(\Delta) \geq \mathbf{0}. \tag{C.1}$$

Let α be the *largest feasible step* that yields:

$$\mathbf{p} + \alpha \Delta \mathbf{p} \geq \mathbf{0}, \tag{C.2}$$

for vectors $\mathbf{p}$ and $\Delta\mathbf{p}$. Then α is obtained by:

$$\alpha = \min_i \alpha_i, \quad \text{where} \quad \alpha_i = \begin{cases} -\dfrac{p_i}{\Delta p_i} & \text{if } \Delta p_i < 0 \\ \infty & \text{otherwise} \end{cases}. \tag{C.3}$$

Use equation (C.3) to determine α_x^t, α_x^n, $\alpha_x^{t,n}$ defined by the following bound conditions:

$$\mathbf{x} + \alpha_x^t \mathbf{s}_x^t \geq \mathbf{0}, \quad \mathbf{x} + \alpha_x^n \mathbf{s}_x^n \geq \mathbf{0}, \quad \mathbf{x} + \alpha_x^{t,n} \Delta\mathbf{x} \geq \mathbf{0}. \tag{C.4}$$

Similarly as before, determine α_z^t, α_z^n, $\alpha_z^{t,n}$ defined by:

$$\mathbf{z} + \alpha_z^t \mathbf{s}_z^t \geq \mathbf{0}, \quad \mathbf{z} + \alpha_z^n \mathbf{s}_z^n \geq \mathbf{0}, \quad \mathbf{z} + \alpha_z^{t,n} \Delta\mathbf{z} \geq \mathbf{0}. \tag{C.5}$$

Let us now define the following auxiliary quantities:

$$\alpha_{\min}^t = \min(\alpha_x^t, \alpha_z^t, \alpha_x^{t,n}, \alpha_z^{t,n}), \tag{C.6}$$

and

$$\alpha_{\min}^n = \min(\alpha_x^n, \alpha_z^n, \alpha_x^{t,n}, \alpha_z^{t,n}), \tag{C.7}$$

which are used to compute the *largest feasible trust-region radius* Δ_k^{f} by:

$$\Delta_k^{\mathrm{f}} = \tau_k \begin{cases} \alpha_{\min}^t \|\mathbf{s}^t\| & \text{if } \alpha_{\min}^t \leq \alpha_{\min}^n \\ \alpha_{\min}^n \|\mathbf{s}^n\| & \text{otherwise} \end{cases}, \tag{C.8}$$

where $\tau_\mu \in (0,1)$ is used to enforce $\mathbf{x}_k(\Delta), \mathbf{z}_k(\Delta) > \mathbf{0}$, defined as:

$$\tau_k = 1 - \min\{10^{-2}, 10^{+2}\mu_k^2\}. \tag{C.9}$$

For efficiency reasons, all the previous calculations should be skipped when:

$$\mathbf{x}_k + \Delta\mathbf{x}_k \geq \mathbf{0} \qquad \text{and} \qquad \mathbf{z}_k + \Delta\mathbf{z}_k \geq \mathbf{0}. \tag{C.10}$$

For this trivial case, we set $\Delta_k^{\mathrm{f}} = \infty$.

C.2 Central Neighbourhood: Largest Central Trust-Region Radius

This section presents the approach for the calculation of the *largest central trust-region radius* Δ_k^{c} that satisfies the central neighbourhood condition:

$$\mathbf{X}_k(\Delta)\mathbf{z}_k(\Delta) \geq \gamma\mu_k(\Delta)\mathbf{e}. \tag{C.11}$$

From equations (3.44) and (3.45), it follows that the following cases are possible:

Case I: $\alpha^n = 1$ and $\alpha^t = \dfrac{\Delta}{\|\mathbf{s}^t\|}$

Compute:

$$\Delta_{\mathrm{I}}^c = \min\{t \in \mathbb{R}^+ : f_i(t) = 0,\ i = 1, \ldots, n\}, \tag{C.12}$$

where $f_i(t)$ is the quadratic function defined by:

$$f_i(t) := \alpha_i t^2 + \beta_i t + \lambda_i, \tag{C.13}$$

with coefficients α_i, β_i and λ_i given by:

$$\alpha_i = b_i d_i n - \gamma \mathbf{b}^T\mathbf{d} \tag{C.14}$$

$$\beta_i = (a_i d_i + b_i c_i)n - \gamma(\mathbf{a}^T\mathbf{d} + \mathbf{b}^T\mathbf{c}) \tag{C.15}$$

$$\lambda_i = a_i c_i n - \gamma \mathbf{a}^T\mathbf{c} \tag{C.16}$$

and auxiliary vectors $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}$ defined as:

$$\mathbf{a} = \mathbf{x} + \mathbf{s}_x^n, \qquad \mathbf{b} = \frac{\mathbf{s}_x^t}{\|\mathbf{s}^t\|}, \tag{C.17}$$

$$\mathbf{c} = \mathbf{z} + \mathbf{s}_z^n, \qquad \mathbf{d} = \frac{\mathbf{s}_z^t}{\|\mathbf{s}^t\|}. \tag{C.18}$$

Case II: $\alpha^n = \dfrac{\Delta}{\|\mathbf{s}^n\|}$ and $\alpha^t = \dfrac{\Delta}{\max(\|\mathbf{s}^n\|, \|\mathbf{s}^t\|)}$

Compute:

$$\Delta_{\mathrm{II}}^c = \min\{t \in \mathbb{R}^+ : f_i(t) = 0,\ i = 1, \ldots, n\}, \tag{C.19}$$

where $f_i(t)$ is the quadratic function defined by (C.13). However, the auxiliary vectors $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}$ are now defined as:

$$\mathbf{a} = \mathbf{x}, \qquad \mathbf{b} = \frac{\mathbf{s}_x^n}{\|\mathbf{s}^n\|} + \frac{\mathbf{s}_x^t}{\max(\|\mathbf{s}^n\|, \|\mathbf{s}^t\|)}, \tag{C.20}$$

$$\mathbf{c} = \mathbf{z}, \qquad \mathbf{d} = \frac{\mathbf{s}_z^n}{\|\mathbf{s}^n\|} + \frac{\mathbf{s}_z^t}{\max(\|\mathbf{s}^n\|, \|\mathbf{s}^t\|)}. \tag{C.21}$$

After computing both Δ_{I}^c and Δ_{II}^c, we set $\Delta_k^c = \min(\Delta_{\mathrm{I}}^c, \Delta_{\mathrm{II}}^c)$.

Note that for efficiency reasons, one can first check if:

$$\mathbf{X}_k(\infty)\mathbf{z}_k(\infty) \geq \gamma\mu(\infty)\mathbf{e}. \tag{C.22}$$

In this trivial case, it follows that $\Delta_k^c = \infty$.